



# **GamBet 4.0 Reference Manual**

PO Box 13595, Albuquerque, NM 87192 U.S.A.

Telephone: +1-505-220-3975

Fax: +1-617-752-9077

E mail: [techinfo@fieldp.com](mailto:techinfo@fieldp.com)

Internet: <http://www.fieldp.com>

## Read this first

With the impressive graphics and computational power of modern computer codes, we tend to forget that they don't always give the right answer. Sources of error may include invalid input parameters, misinterpretation of results and program bugs. Computer "simulations" should not be considered the absolute arbiter in technical applications. Cross checks and careful analysis are essential. This is particularly important in radiation transport calculations that could affect personnel safety.

- **GamBet** can be a useful aid in the design of radiation shielding, but the simulations alone are insufficient. The results must be combined with analytic estimates, practical guidelines and careful monitoring to ensure safety.
- Although **GamBet** is well suited to research investigations of new approaches to radiation therapy, the code was not designed for planning radiation treatments in a clinical setting. **GamBet is not certified or recommended for use in clinical radiation therapy.**

Although we make all possible efforts to improve and to correct **GamBet**, it is impossible to guarantee that a code of its complexity is absolutely free of error. The user is responsible for checking the validity and inputs and results. Field Precision assumes no liability for applications of **GamBet** beyond a refund of the purchase price.

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>GBLaunch</b>	<b>8</b>
2.1	Setting the program and data directories . . . . .	8
2.2	Creating and running tasks . . . . .	8
2.3	Tools . . . . .	10
<b>3</b>	<b>Simulation modes</b>	<b>12</b>
<b>4</b>	<b>Structure of the GamBet control script</b>	<b>14</b>
<b>5</b>	<b>Using the setup dialog to create a script</b>	<b>16</b>
<b>6</b>	<b>Geometry commands</b>	<b>22</b>
6.1	Loading the geometry file . . . . .	23
6.2	Loading and modifying electric field information . . . . .	23
6.3	Loading and modifying magnetic field information . . . . .	24
<b>7</b>	<b>Composition commands</b>	<b>26</b>
<b>8</b>	<b>Source commands</b>	<b>29</b>
8.1	Particle input from source files . . . . .	30
8.2	Particle input from <b>Trak</b> and <b>OmniTrak</b> . . . . .	31
8.3	Number of showers and source properties . . . . .	32
<b>9</b>	<b>Process commands</b>	<b>34</b>
9.1	Penelope control . . . . .	34
9.2	Variance reduction control . . . . .	36
9.3	Simulation control . . . . .	37
<b>10</b>	<b>Running GamBet: interactive and background modes</b>	<b>41</b>
<b>11</b>	<b>GamBet: parallel operation</b>	<b>44</b>
<b>12</b>	<b>Input/output files</b>	<b>48</b>
<b>13</b>	<b>Contents of the listing file</b>	<b>51</b>
13.1	Source information . . . . .	51
13.2	Penelope setup information . . . . .	51
13.3	Primary particle statistics . . . . .	52
13.4	Energy conservation . . . . .	53
13.5	Dose assignment statistics . . . . .	53
13.6	Particle flux statistics . . . . .	54

<b>14 GBView2</b>	<b>55</b>
14.1 File menu . . . . .	55
14.2 Spatial plots menu . . . . .	56
14.3 Dose analysis menu . . . . .	59
14.4 Scan plot menu . . . . .	61
<b>15 GBView3</b>	<b>63</b>
15.1 File menu . . . . .	63
15.2 Dose analysis menu . . . . .	64
15.3 Plane plots . . . . .	66
15.4 Slice plots . . . . .	68
15.5 Analysis in a slice . . . . .	70
15.6 Surface plots . . . . .	71
15.7 Saving and loading views . . . . .	72
<b>16 GBView2 and GBView3 script operation</b>	<b>74</b>
<b>17 Penelope predefined materials</b>	<b>76</b>
17.1 Elements . . . . .	76
17.2 Compounds and mixtures . . . . .	77
<b>18 Penelope data files</b>	<b>82</b>

---

# 1 Introduction

**GamBet** (gamma and beta particles) is a comprehensive software suite to simulate the transport of energetic electrons, photons and positrons through matter over the energy range from 50 eV to 1 GeV. The program tracks the history of primary and secondary particles produced by material interactions by the Monte Carlo method. **GamBet** is an essential tool for applications such as X-ray imaging, radiation detectors, electron-beam generation of bremsstrahlung radiation and research on radiation medical treatments. **GamBet** may be employed as an extension to the **Trak** and **OmniTrak** charged-particle codes or as a stand-alone Monte Carlo simulator. The program produces files of power deposition that may be ported to **TDiff** or **HeatWave** for thermal analyses of radiation effects.

The program `gambet.exe` combines Field Precision technology developed for finite-element codes with the **Penelope** radiation-physics package<sup>1</sup>. The **Penelope** routines perform the Monte Carlo tasks in **GamBet**: generation of atomic cross sections, prediction of single-particle interactions with matter and the creation of secondary particles. The Field Precision routines handle run control and all matters related to the division of space. Tasks include the organization of input/output data for large distributions of particles, calculation of statistics, identification of material boundaries, generation of escape-particle records, variance reduction techniques, creation of records of spatial variations of dose and control of parallel operation. Other programs of the **GamBet** package address mesh generation and interactive analysis of results. The basic **GamBet** suite includes the components listed in Table 1. Table 2 lists the Field Precision programs that produce compatible input for **GamBet**.

**GamBet** employs a unique approach to Monte Carlo calculations with several advantages over other programs:

1. Conformal meshes are used for the division of the material volume, giving high-accuracy representations of slanted or curved boundaries.
2. The effects of arbitrary 2D and/or 3D electric and magnetic fields may be included.
3. Sophisticated graphical pre- and post-processors are available for mesh generation and the analysis of dose distributions and particle orbits.
4. The program accepts direct input of electron or positron distributions from the **Trak** and **OmniTrak** codes.
5. Dose quantities are recorded on the conformal mesh; therefore, each element acts as a subzone. The user can make analysis decisions in the post-processing stage, reducing the need to re-run simulations.
6. Dose distribution files may be ported directly to **TDiff** and **HeatWave** for static or dynamic thermal calculations.

---

<sup>1</sup> Francesc Salvat, Jos M. Fernández-Varea, and Josep Sempau, **PENELOPE-2006: A Code System for Monte Carlo Simulation of Electron and Photon Transport**, (Proceedings of a Workshop/Training Course, July 4-7, 2006), NEA Number 6222.

Table 1: Components of the **GamBet** code suite

<b>Program</b>	<b>Function</b>
<code>mesh.exe</code>	Definition of 2D geometries, preparation of conformal triangular meshes.
<code>geometer.exe</code>	Interactive definition of 3D geometries
<code>metamesh.exe</code>	Creation of 3D conformal hexahedron meshes from geometric information
<code>gambet.exe</code>	Main technical program for Monte Carlo simulations
<code>gambetmp.exe</code>	A background version of the technical program called for runs with parallel processing.
<code>gbview2.exe</code>	Interactive post-processor for plots and analysis of 2D dose files and trajectory records
<code>gbview3.exe</code>	Interactive post-processor for plots and analysis of 3D dose files and trajectory records
<code>gendist.exe</code>	Utility to create or to analyze large input particle distributions

Table 2: Field Precision programs that produce input files compatible with **GamBet**

<b>Program</b>	<b>Function</b>
<code>estat.exe</code>	2D electric field solutions
<code>permag.exe</code>	2D magnetic field solutions
<code>trak.exe</code>	Simulation of 2D electron guns and transport devices, creation of input particle distributions
<code>hiphi.exe</code>	3D electric field solutions
<code>magnum.exe</code>	3D magnetic field solutions
<code>omnitrak.exe</code>	Simulation of 3D electron guns and transport devices, creation of input particle distributions

7. Particle output files may be used as input for subsequent runs or ported to **Trak** and **OmniTrak**.
8. Advanced variance-reduction techniques are employed, including interaction forcing and splitting/reduction of particles that enter specified regions of the solution volume.
9. The code supports efficient parallel processing on multi-core Windows computers.

To install the **GamBet** package, download and run `gambetsetup.exe`. The installation utility creates a new directory and then adds programs, instruction manuals, material data files and examples. The executables and the instruction manuals are located in the directory `C:\GAMBET`. The Penelope data files are in the directory `C:\GAMBET\REFERENCE`. The installer also adds a shortcut to `gblaunch.exe` on the desktop.

The programs must be activated before they can be used. Instructions are supplied with the package. Note that activation is independent of installation. You can update the programs without affecting activation of the computer. To avoid update problems, we recommended that you do not store data and other files to the program directory `C:\GAMBET`.

---

## 2 GBLaunch

The first step to run any of the programs in the **GamBet** package is to set up `gblaunch.exe`, the program controller. The utility helps you to organize your work in two ways:

- Only one shortcut is required on your desktop, minimizing clutter.
- All programs open in a specified data directory, eliminating redundant trips through the directory tree.

### 2.1 Setting the program and data directories

Figure 1 shows the screen display of the **GBLaunch** controller. To begin, click the *Program folder* button in the *Control* group. In the dialog move to the directory that contains the executables and click *OK*. The default location is `\fieldp\gambet`. The buttons in the technical group should all become active. An inactive button indicates a missing executable. Setting the program directory also sets a registry entry so that **GamBet** can locate the folder that contains Penelope data files.

It is a good practice is to collect all input and output files for related **GamBet** solutions in a specific data directory. Use the *Data Folder* button to set the current location. Programs that are opened after the change will read and write to the directory. (Note that the setting does not affect previously-opened programs.) The *Information* area at the bottom of the control box shows: 1) the current data directory, 2) the last operation performed and 3) the program directory.

### 2.2 Creating and running tasks

**GamBet** programs are optimized for the latest generation of multi-core or multi-processor PCs. The Professional solution programs (**MetaMesh**, **GamBet**,) feature true parallel operation. A second feature in both the Basic and Professional programs is the capability to run multiple independent calculations simultaneously. All solution programs can run in the background if launched from a Windows batch file. Background operation is automatic and faster than running in a window. The *Create task* and *Run task* commands in **GamBet** make it easy to use batch files. With the commands you can 1) quickly define multi-step calculations (*tasks*) in an interactive dialog, 2) launch simultaneous tasks in the background and 3) find out which tasks are running.

The *Create task* button calls up the dialog of Figure 2. Supply a file prefix `FPREFIX` that indicates the function of the task. The task information will be stored in a DOS batch file `FPREFIX.BAT` created in the current **GamBet** data directory. Commands in the file are compatible with all recent Windows versions including Windows 7. Each row represents an operation (batch file command). The first column defines the action. Clicking on a cell brings up a menu that includes all **GamBet** programs capable of background operation. In addition, several relatively safe DOS commands are included (`ERASE`, `COPY`, `MOVE`, `RENAME` and `REM`). All commands operate on a file (*FileIn* column). The DOS commands `COPY`, `MOVE` and `RENAME`

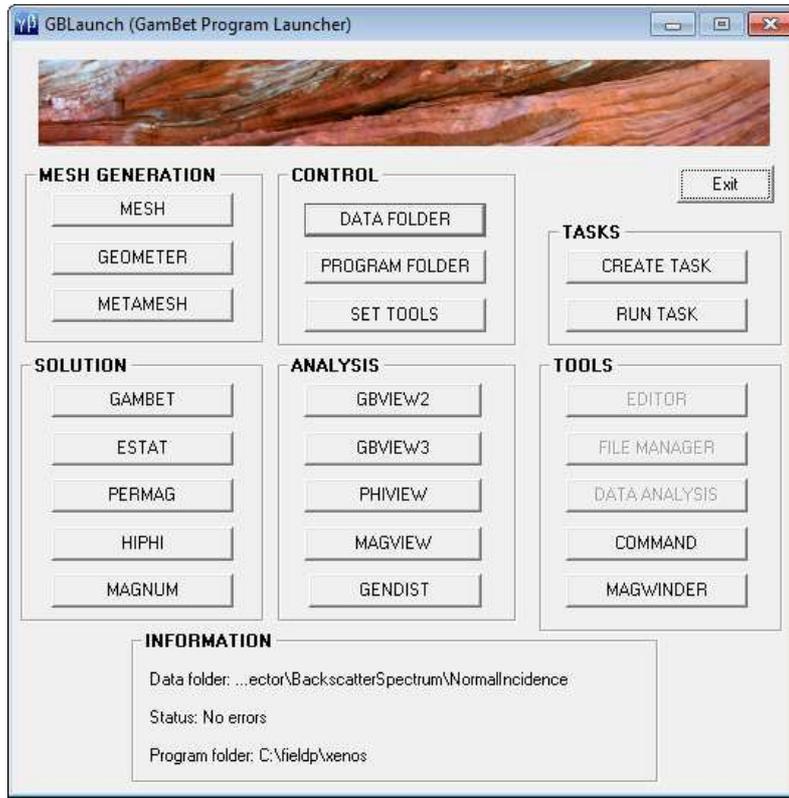


Figure 1: GamBet program launcher

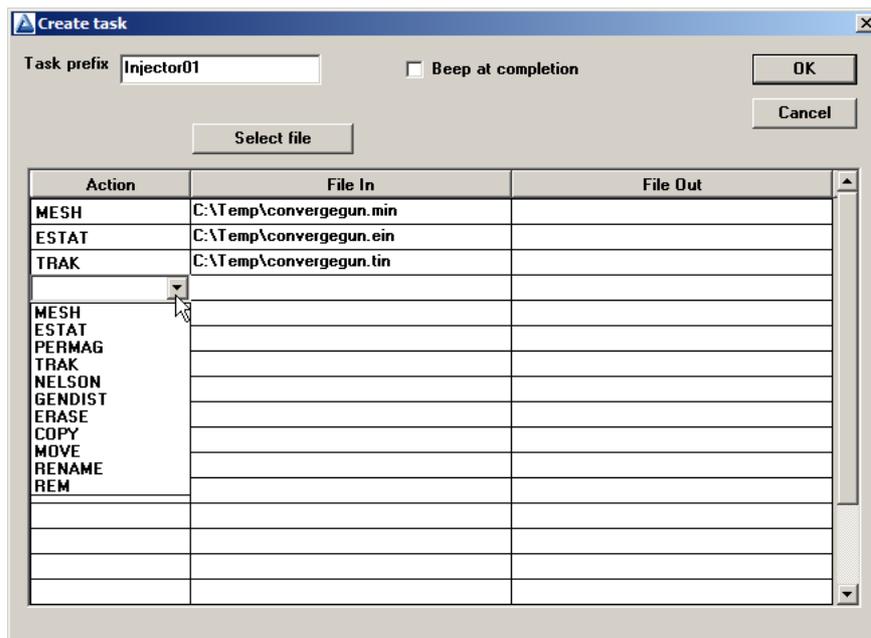


Figure 2: Create task dialog

require a second file name (*FileOut* column). You can type file names in the cells. By default, the files are in the **GamBet** working directory, but you can include path information if the files are in other directories. Alternatively, you can click in a cell and then pick the *Select file* command to use the standard Windows dialog for choosing files anywhere on the computer. For the **GamBet** programs, the dialog displays only files with appropriate suffixes (*e.g.*, \*.GIN for **E**).

Click the *OK* button when the sequence is complete to create the batch file. Here is an example:

```
REM GamBet batch file, Field Precision
START /B /WAIT C:\fieldp\gambet\mesh64.exe C:\Temp\convergegun
START /B /WAIT C:\fieldp\gambet\gambet64.exe C:\Temp\convergegun
START /B /WAIT C:\fieldp\gambet\gbview264.exe C:\Temp\convergegun
ERASE *.?1s
START /B /WAIT C:\fieldp\gambet\notify.exe
IF EXIST Electrode01.ACTIVE ERASE Electrode01.ACTIVE
```

The operations listed perform a complete **GamBet** calculation in the background and then erase all listing files. The example has some notable features:

- The operations are performed sequentially because data from one operation may be used in the next. To run calculations in parallel, define and run multiple tasks.
- You can modify the file with an editor if you are familiar with DOS commands.
- The DOS commands recognize the standard wildcard conventions (\* for any character grouping, ? for any character).
- The programs adds the command `notify.exe` to the task sequence if *Audio alarm* was checked. In this case, the computer beeps when a task is completed.
- The final command to erase a file `FPREFIX.ACTIVE` is added to all batch files. The presence of the file indicates that the task is running.

Click the *Run task* button when you have created tasks or moved predefined task files to the data directory. The dialog (Figure 3) organizes tasks into two groups: ones that are available to run and ones that are currently running (*i.e.*, `FPREFIX.ACTIVE` has been detected). To launch a task, choose one from the top list and click *OK*. The program creates a file `FPREFIX.ACTIVE` and runs the batch file. The program sequence runs silently in the background. In the meantime, you can prepare other inputs or run other tasks.

## 2.3 Tools

You will probably use additional software tools working with **GamBet** and other technical programs. You can set **GamBet** to launch your favorite utility programs. To define a text editor, press the *Setup tools* button in the *Control* section. In the dialog, choose the *Editor* option. Then, move to the appropriate directory and select the program. When you exit the procedure, the *Editor* button in the *Tools* section becomes active.



Figure 3: Run task dialog

You can use the *Set tools* command to define two other utilities. For the *File manager* option, select a program like **Windows Explorer** or **FreeCommander**. Use the *Data analysis* button for a spreadsheet, plotting program or mathematical analysis software. The *textitCommand* button opens a DOS window if you want to run programs from the command prompt or under the control of your own batch files.

---

### 3 Simulation modes

There are three available modes for a **GamBet** simulation:

- SingleParticle
- ContinuousBeam
- PulsedBeam

The mode determines units of data quantities recorded in the output files. **GamBet** sets the mode depending on entries in the source or particle input lists and the presence of the **TPULSE** command (see Table 3). The single-particle mode corresponds to a standard Monte Carlo run where quantities are weighted according to the number of input particles. In the continuous-beam mode, a current or flux is associated with each model particle. In this case, **GamBet** can calculate dose rates and particle flux between regions. In the pulsed-beam mode, the model particle current is active for a specified time. In this case, the code calculates dose and integrated flux (total number).

Table 3: **GamBet** simulation modes

Mode	Current/flux entry in PLIST or SLIST data	TPULSE specified
SingleParticle	No	(No effect)
ContinuousBeam	Yes	No
PulsedBeam	Yes	Yes

The units of deposited dose (Table 4) depend on the calculation mode. The standard SI dose unit is the gray ( $1 \text{ Gy} = 1 \text{ J/kg} = 100 \text{ rads}$ ). The following considerations apply to the calculation of deposited dose in planar 2D calculations.

- In the single-particle mode, each incident primary particle is treated as one particle/m uniformly distributed in  $z$ .
- In the continuous-beam and pulsed-beam modes, the flux value for a primary particle is in A/m for electrons and positrons and particles/s/m for photons. The flux is assumed to be uniformly distributed in  $z$ .

The unit of particle flux between regions is particles/primary for all calculation modes and symmetries. Table 5 shows units of energy flux between regions.

Table 4: Recorded dose quantities

<b>Mode</b>	<b>Recorded quantity</b>
SingleParticle	eV/kg/primary
ContinuousBeam	J/kg/s (Gy/s)
PulsedBeam	J/kg (Gy)

Table 5: Recorded energy-flux quantities

<b>Mode</b>	<b>Symmetry</b>	<b>Recorded quantity</b>
SingleParticle	Planar2D,Cylindrical2D,3D	eV/primary
ContinuousBeam	Planar2D	J/s/m
	Cylindrical2D,3D	J/s
PulsedBeam	Planar2D	J/m
	Cylindrical2D,3D	J

---

## 4 Structure of the GamBet control script

The **GamBet** script is a text file with data lines containing commands and parameters. The program makes no distinction between upper and lower case. Entries on a line may be separated by the following delimiters:

- Space
- Comma [,]
- Tab character
- Colon [:]
- Equal sign [=]
- Left parenthesis [(
- Right parenthesis [)]

You may use any number of delimiters in a line. This means that you can add indentations and customize the appearance of the script. **GamBet** ignores blank lines and comment lines. Comment lines begin with the symbol '\*' (asterisk).

Most parameters are real numbers. The following formats are valid.

- 1.000
- 5.67E6
- 6.8845E+09
- 5

The final number is interpreted as 5.0.

The **GamBet** script has the following organization:

```
GEOMETRY
  (Geometry commands)
END

COMPOSITION
  (Composition commands)
END

SOURCE
  (Source commands)
END

PROCESS
  (Process commands)
END
```

The four sections must appear in the order shown. Allowed commands may be placed in any order within a section. **GamBet** reads and analyzes all commands of a section before proceeding with calculations. The exception is the **DUnit** command for coordinate conversion. The command affects only subsequent commands.

The script sections have the following functions:

**GEOMETRY** Load a 2D or 3D mesh that defines the material geometry. Optionally, you can also load **TriComp** or **AMaze** solution files to define applied electric and magnetic fields.

**COMPOSITION** Define material properties for Penelope calculations and associate materials with the regions of the geometric mesh.

**SOURCE** Specify the properties of incident particles (electrons, photons and positrons).

**PROCESS** Control the Monte Carlo calculation.

The following example illustrates a simple **GamBet** script:

```
* File: COPPERSHEET.GIN
GEOMETRY
  DUnit = 1.0E6
  GFile2D = CopperSheet.MOU (Rect)
END
COMPOSITION
  Material = 29
  Region(1) = 1
END
SOURCE
  SList
    E 20.0E6 0.00 0.00 0.00 1.00 0.00 0.00
  End
  NPMult = 100000
END
PROCESS
  EAbs(Electron) = 1.0E5
  EAbs(Photon) = 1.0E5
  EAbs(Positron) = 1.0E5
  DsMax(1) = 0.20
  PlotOn(1)
END
ENDFILE
```

---

## 5 Using the setup dialog to create a script

The setup dialog is a quick and easy way to construct basic **GamBet** control scripts. The interactive environment helps you remember commands and syntax rules. You can use the dialog to create a basic structure and then add details or modify parameters with a text editor. This chapter briefly summarizes the function of fields in the dialog. Chapters 6 through 9 give detailed information on standard and advanced commands.

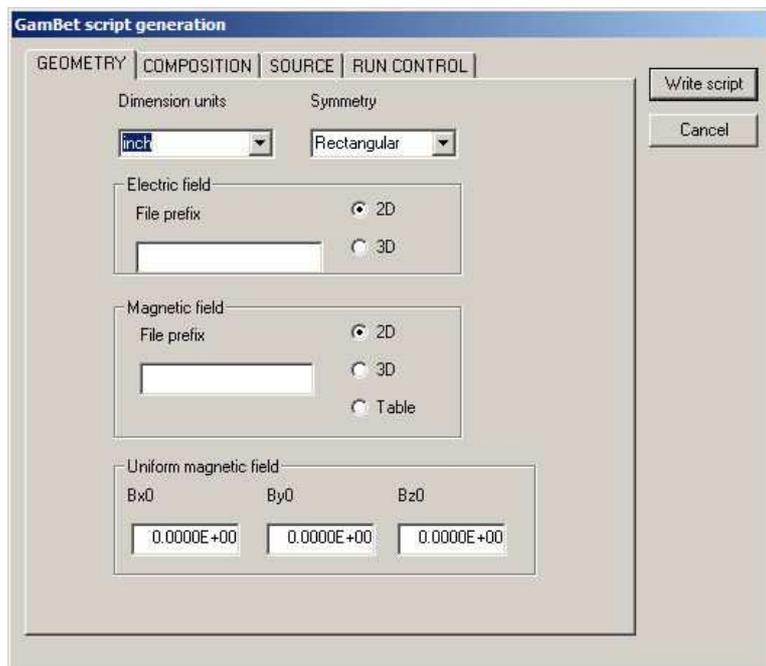


Figure 4: Setup dialog, Geometry tab

**GamBet** calculations require a geometry mesh. A two-dimensional mesh file (`FPREFIX.MOU`) or a three-dimensional file (`FPREFIX.MDF`) must be available to use the setup dialog. In the main menu, click on the **Setup** command or tool. In the file selection dialog, choose the input mesh file. **GamBet** checks the number of regions in the mesh and reads the region names if they are available. If the file is a valid mesh, the dialog of Fig. 4 appears. Note the tabs at the top of window: there is one tab for each of the command groups in the script (Chap. 4).

The *Geometry* tab is active in Figure 4. The following fields apply to all scripts.

### Dimension units

This entry informs **GamBet** of the length units used in mesh file. Click the down-arrow to display the full list of options. Choose *Custom* if the desired unit is not included in the list. In this case, the program makes a dummy entry in the script that you can complete with a text editor.

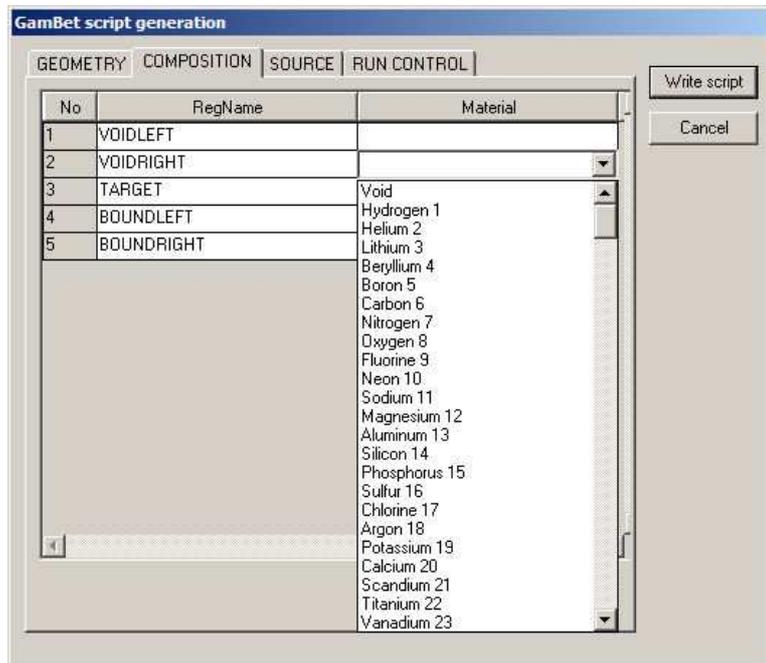


Figure 5: Setup dialog, *Composition* tab

### Symmetry

This field is active only for two-dimensional meshes. The *Rectangular* option implies that there are variations in  $x$  and  $y$  with infinite length along  $z$ . For the *Cylindrical* option, the system has azimuthal symmetry with variations along  $r$  and  $z$ .

Enter values in fields of the electric and magnetic field groups if you want to load information from output files generated by **EStat**, **HiPhi**, **PerMag** or **Magnum**. Electrostatic and magnetostatic fields act on electrons and positrons in the calculation.

### File prefix

Enter the prefix of the field data file. The string must be from 1 to 46 characters in length. Depending on the setting of 2D/3D radio buttons, the program adds the appropriate suffix. For electric fields, the resulting file names are **FPREFIX.EOU** (**EStat**, 2D) and **FPREFIX.HOU** (**HiPhi**, 3D). The file names for magnetic fields are **FPREFIX.POU** (**PerMag**, 2D) and **FPREFIX.MOU** (**Magnum**, 3D). For the *Table* option in the magnetic field group, the resulting file name is **FPREFIX.TAB**.

### Bx0, By0, Bz0

Define uniform values of magnetic field along the  $x$ ,  $y$  and  $z$  direction. Enter component values in tesla.

Figure 5 shows the display for the second tab. Values set the the material composition of regions in the simulation volume. The grid has one row for each region of the mesh. The columns of the grid show the region number, the region name (if available) and the material composition. To define a material, move the mouse into a *Material* box and click the left button.

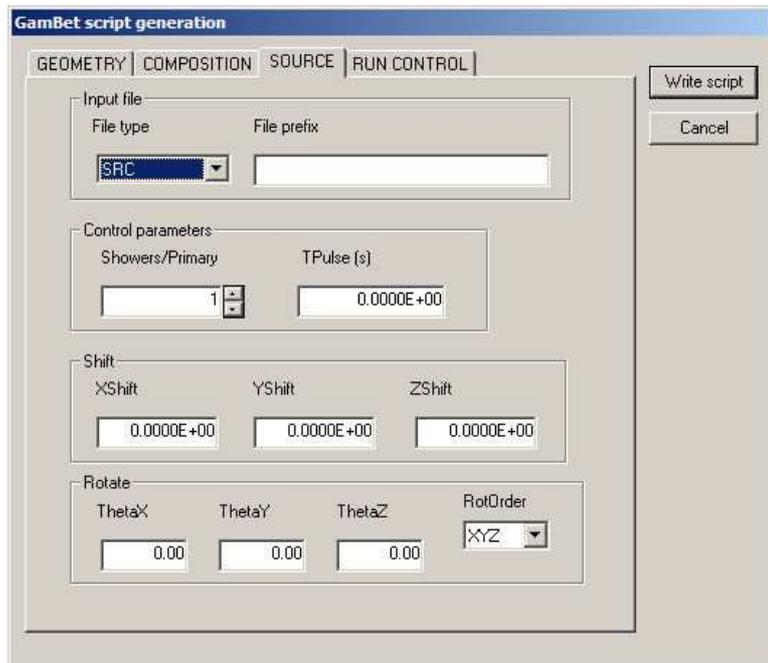


Figure 6: Setup dialog, *Source* tab

The drop-down list shows the full set of Penelope materials (Chap. 17) and the additional choice of *Void* (vacuum). Scroll through the list to find an appropriate material and then click the left mouse button to set the composition for the box. Note that the Penelope numbers initially follow the atomic number of elements. After that, compounds are listed roughly alphabetically. Chapter 17 summarizes the choices. Each box in the column must be specified. The program issues an error message if there is a missing definition for a region.

Fields in the *Source* tab define the properties of injected particles. The following quantities must be defined for each primary particle: type (electron, photon or positron), kinetic energy, position and direction. The quantities are contained in a list with one data line for each primary particle. The list may be either incorporated in the script or imported from a separate file. There are two list formats:

- The native **GamBet** format, where the list file has a name of the form `FPREFIX.SRC` (Sect. 8.1).
- The input/output format of **Trak** and **OmniTrak**, where the list file has a name of the form `FPREFIX.PRT` (Sect. 8.2).

The following two fields appear in the *Input file* group.

### File type

Specifies whether input will be in the *SRC* or *PRT* format.

### File prefix

The prefix of a file (FPREFIX.SRC or FPREFIX.PRT) of particle data lines available in the working directory. Sections 8.1 and 8.2 describe the file formats. If there is no entry in this file, **GamBet** creates a dummy entry in the script of the form:

```
SList
*   Type   Kinetic   x   y   z   ux   uy   uz   [Current/flux]
*           energy
*   =====
*   (Add particle parameters)
End
```

In this case, you must add particle data lines with a text editor.

The *Control parameters* group has the following entries:

### Showers/primary

Set the parameter  $NPMult$ , the number of showers to initiate per primary particle. Larger values give improved statistics. Regarding the assignment of deposited dose, if a primary particle carries current  $I$ , then the current per shower particle is  $I/N_{pmult}$ .

### TPulse

Set the pulselength  $\Delta t_p$  for the input beam of primary particles. Enter the value in seconds. This parameter is active only if current or flux is associated with the primary particles. If a value is supplied, **GamBet** records accumulated dose in the output file. If  $\Delta t_p$  is not defined, the program records the dose rate.

The entries in the final two groups (*Shift* and *Rotation*) may be used to apply global translations or rotations to the primary particles.

### XShift, YShift, ZShift

Shift the initial positions of particles relative to the geometry mesh. Values are modified according to  $x' = x + X_{shift}, \dots$ . Enter values in units set by the specified dimension units. Note that rotations are applied before shifts.

### XRot, YRot, ZRot, RotOrder

Rotate the particle direction vectors relative to the geometry mesh. Enter the values of angles in degrees. The optional string parameter *RotOrder* specifies the order of rotations. Rotations are applied before shifts.

Values in fields of the *Run* control tab of Fig. 7 control the Monte Carlo calculation.

### Maximum energy

The maximum energy  $E_{max}$  for the calculation of cross sections by Penelope. Enter the value

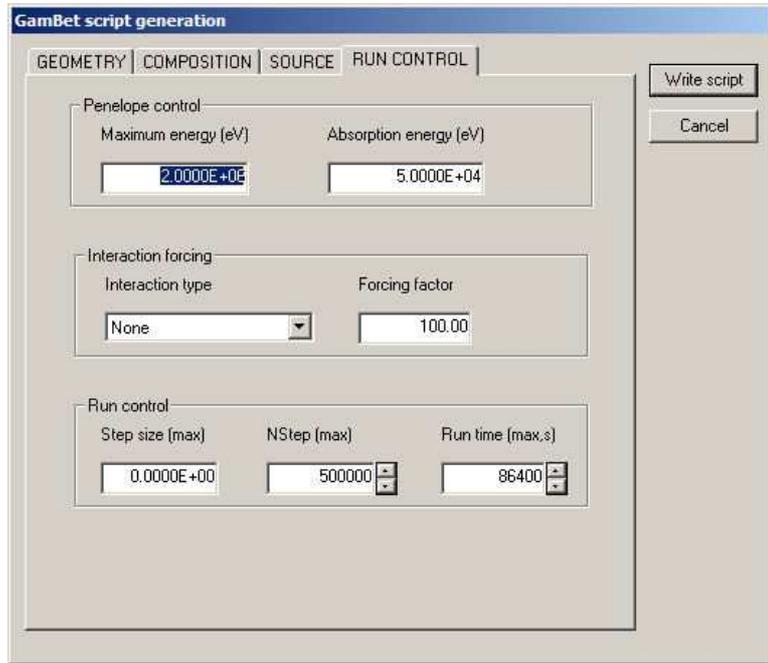


Figure 7: Setup dialog, *Process* tab

in eV (electron volts). Usually,  $E_{max}$  is slightly higher than the maximum energy of primary particles. In a solution with electric fields be sure to include the possibility of electron or positron acceleration.

### Absorption energy

A particle is absorbed in a material region when its energy drops below a value  $E_{abs}$ . In this case, the remaining particle kinetic energy is deposited in the currently-occupied element. The value of  $E_{abs}$  must be greater than or equal to 50.0 eV, the lower limit in the Penelope tables. Higher values substantially reduce the run time.

Interaction forcing improves statistics for low-probability events. The program adjusts particle weights and interaction probabilities to maintain unbiased statistics.

### Interaction type

The two choices are bremsstrahlung and pair production. The first option is useful for modeling X-ray targets while the second option may be applied to positron sources.

### Forcing factor

Enter a real number greater than 1.0, typically 10.0 to 100.0.

The fields in the last group control aspects of the Monte Carlo simulation.

**Step size**

This field is the maximum allowed transit distance for particles per event step. Enter a value in the specified dimension units. The maximum step size should be about 1/10 of the dimension of the smallest material region.

**NStep**

The maximum number of event steps in a shower. The purpose of the limit is to prevent infinite calculations (*i.e.*, electron in a void trapped by a magnetic field). The default value is 500,000. Increase the number if particles stop early. You can check for the condition by inspecting the energy conservation statistics in the listing file (GLS).

**Run time**

The maximum time for the calculation. Enter the integer value in seconds. **GamBet** stops at the maximum run time and records data for the completed showers. This value may be useful for long runs under batch file control.

---

## 6 Geometry commands

COMMAND SUMMARY
-----------------

DUNIT DUnit
GFILE2D FileName [Rect, Cylin]
GFILE3D FileName
EFILE2D FileName [EMult]
EFILE2D FileName BTHETA [EMult] [BBMult]
EFILE3D FileName [EMult]
BFILE2D FileName [BMult]
BFILE3D FileName [BMult]
BTABLE: FileName [Zoffset,Zult, BOffset, BMult]
BUNI: Bx0 By0 Bz0
SHIFT [E,B] XShift YShift ZShift
ROTATE [E,B] RotX, Roty, Rotz

**GamBet** can employ geometric and field information from three independent conformal meshes.

**GFILE** [Required] Geometry of materials used for Penelope operations.

**EFILE** [Optional] Applied electric field.

**BFILE** [Optional] Applied magnetic field.

You can combine 2D and 3D meshes in a single simulation. You can also combine 2D meshes having different symmetries (planar or cylindrical). The output dose file has the same dimensionality and symmetry as the **GFILE**.

The following considerations apply when you combine meshes.

- By default the origins of the meshes are coincident. Here the origin is the point where  $(x, y, z) = (0, 0, 0)$ . The **GFILE** defines the reference coordinate system. Use the **SHIFT** command to translate the **EFILE** or **BFILE** meshes.
- By default, the coordinate axes are colinear. A cylindrical mesh has  $z$  along the  $z$  direction in 3D space and  $r$  lies in the  $x$ - $y$  plane. A planar mesh has variations in  $x$ - $y$  and infinite length along  $z$ . Use the **ROTATE** command to rotate the **EFILE** and **BFILE** meshes relative to the **GFILE** mesh.
- If the **GFILE** is two-dimensional, the output dose distributions have two-dimensional symmetry, even if the particle distribution in the simulation is not symmetric. Non-symmetric particle distributions may be specified in the **SOURCE** section or could result from motion in a field with different symmetry. The dose in a file with planar symmetry is integrated along  $z$ ; therefore, only the  $x$ - $y$  variations are displayed. In a cylindrical file, energy is assigned to  $r$ - $z$  elements independent of the model particle azimuth.

## DUNIT Unit

Enter a conversion factor for coordinates that appear in subsequent commands. The conversion also applies to coordinates in the **GFILE** for input from **Mesh** and **MetaMesh**. In this case, ensure that the **DUNIT** command appears before the **GFILE2D** or **GFILE3D** commands. The **Unit** argument may be one of the following strings for common units: **angstrom**, **nanometer**, **micrometer**, **mil**, **mm**, **cm**, **inch**, **foot**, **yard**, **meter**, **kilometer** or **mile**. For custom units, enter a real number equal to the number of **Mesh** or **MetaMesh** units per meter. The quantity is the number of length units per meter. For example, if coordinates in the **GFILE** were specified in microns, set  $DUnit = 1.0 \times 10^6$  and enter coordinates in subsequent commands in microns.

## 6.1 Loading the geometry file

### **GFILE2D FileName [Rect, Cylin]**

Define the material geometry for the Monte Carlo calculation. The command requires an output file from **Mesh** (**FileName.MOU**) or the **TriComp** solution programs **EStat** (**FileName.EOU**), **PerMag** (**FileName.POU**) or **Pulse** (**FileName.POU**). The file must be available in the working directory. The keywords **RECT** or **CYLIN** must be included when loading a file from the 2D **Mesh** program. In the **RECT** option, the program interprets the geometry as variable in  $x$ - $y$  and infinite in  $z$ . In the **CYLIN** option, the program interprets the geometry as varying in  $r$ - $z$  with symmetry along  $\theta$ . When loading information from a field solution file, **Gambet** extracts the mesh information and reads a file parameter to determine the symmetry.

### **GFILE3D FileName**

Define the material geometry used for the Monte Carlo calculation. The command requires an output file from **MetaMesh** (**FileName.MDF**). The program interprets the geometry as varying in  $x$ - $y$ - $z$ .

## 6.2 Loading and modifying electric field information

### **EFILE2D FileName [EMult]**

Load an electric field solution from **EStat**. Note that the mesh need not have the same symmetry or cover the same volume as the **GFile**. The electric field equals zero at points outside the **EFile** boundaries. The optional parameter **EMult** is a field scaling factor. Values of electrostatic potential are multiplied by **EMult** as they are loaded. For example, a value **EMult** = 0.5 reduces electric field components by a factor of 2.0.

### **EFILE2D FileName BTHETA [EMult] [BBMult]**

The special form of the command loads an electric field solution created with the **EDUMP** command by the **Trak** code running in the **RELBEAM** mode. **Trak** has been modified to record values of the beam-generated toroidal field ( $B_\theta$ ) at nodes in addition to the electrostatic potential. The extra information will be loaded into **Gambet** in response to the keyword **BTHETA**. In this

case, electron orbits will include the effect of the toroidal magnetic field. Values of  $\phi$  and  $B_\theta$  may be scaled with the optional parameters `EMult` and `BBMult`.

### **EFILE3D FileName [EMult]**

Load an electric field solution from `HiPhi`. Note that the mesh need not have the same symmetry or cover the same volume as the `GFile`. The electric field equals zero at points outside the `EFile` boundaries. The optional parameter `EMult` is a field scaling factor.

### **ROTATE E RotEx, RotEy, RotEz [RotOrder]**

Rotate the electric field solution relative to the geometry mesh. Enter the values of angles in degrees. The optional string parameter `RotOrder` specifies the order of rotations (*i.e.*, 'ZX', 'ZYX'). Only the operations specified in `RotOrder` are performed. The default is `RotOrder = 'XYZ'`. Note that rotations are applied before translations.

### **SHIFT E XEShift YEShift ZEShift**

Shift coordinates of the electric field solution relative to the geometry mesh. Enter values in units set by `DUnit`. Note that rotations are applied before translations.

## **6.3 Loading and modifying magnetic field information**

### **BFILE2D FileName [BMult]**

Load a magnetic field solution from `PerMag` or `Pulse`. Note that the mesh need not have the same symmetry or cover the same volume as the `GFile`. The magnetic field equals zero at points outside the `BFile` boundaries. The optional parameter `BMult` is a field scaling factor.

### **BFILE3D FileName [BMult]**

Load a magnetic field solution from `Magnum`. Note that the mesh need not have the same symmetry or cover the same volume as the `GFile`. The magnetic field equals zero at points outside the `BFile` boundaries. The optional parameter `BMult` is a field scaling factor.

### **BTABLE: FileName [ZOffset,ZMult, BOffSet, BMult]**

Determine an applied solenoid-type magnetic field from a table of values along the  $z$  axis at  $r = 0.0$ . The field must have cylindrical symmetry. The parameter `FileName` is the full name of a tabular file. The file contains up to 256 data lines of the form:

```
z    Bz(0,z)
```

The file terminates with an `EndFile` command. The table may also contain comment and blank lines. The positions  $z$  should be specified in units set by the current setting of `DUnit`. Supply values of  $B_z$  in tesla, or you can use the `BMult` parameter to convert from other units. For example, if the data are given in gauss, use `BMult= 1.0 × 10-4`. The following illustrates a table

```

* Table from SOL_LENS.BOU
*      z          Bz
* =====
-9.000E+00  4.421E-06
-8.852E+00  5.926E-05
-8.703E+00  1.173E-04
-8.555E+00  1.780E-04
...
 9.258E+00  1.014E-04
 9.406E+00  7.838E-05
 9.555E+00  5.670E-05
 9.703E+00  3.685E-05
 9.852E+00  1.767E-05
1.000E+01  7.111E-11
ENDFILE

```

Optionally, you can supply four real numbers as parameters in the command: `ZOffSet`, `ZMult`, `BOffSet` and `BMult`. The parameters modify table values entered into the program according to

```

zprog = ZMult*ztab + ZOffSet,
Bzprog = BMult*Bztab + BOffSet

```

A cubic spline routine is used to interpolate values of  $B_z$  along the axis and numerical derivatives are used to calculate off-axis values  $B_r(r, z)$  and  $B_z(r, z)$ . As a result, estimates of  $B_z$  are limited to terms of order  $r^2$ . This means that variations of  $B_r$  are limited to terms of order  $r$  (linear variation).

### **BUNI: Bx0 By0 Bz0**

Set uniform components of the magnetic field. Enter the values in tesla. This command may appear in conjunction with `BFile` or `BTable` commands. In this case, the magnetic field is the sum of all sources.

### **ROTATE B RotBX, RotBy, RotBz**

Rotate the magnetic field solution relative to the geometry mesh. Enter the values of angles in degrees. The optional string parameter `RotOrder` specifies the order of rotations (*i.e.*, 'ZX', 'ZYX'). Only the operations specified in `RotOrder` are performed. The default is `RotOrder = 'XYZ'`. Note that rotations are applied before translations.

### **SHIFT B XBSHIFT YBSHIFT ZBSHIFT**

Shift coordinates of the magnetic field solution relative to the geometry mesh. Enter values in units set by `DUnit`. Note that rotations are applied before translations.

---

## 7 Composition commands

COMMAND SUMMARY
-----------------

<pre>MATERIAL MatNo [CONDUCTOR,INSULATOR] MATERIAL   NAME   COMPONENT [Symbol or atomic number] [Fraction]   DENSITY   CONDUCTOR END REGION RegNo MatNo REGION RegNo VOID REGION RegNo EXCLUDE ECALC RegNo [ON, OFF] BCALC RegNo [ON, OFF] FULLINFO</pre>
---

Commands of the *Composition* section create the temporary file `GAMBET.MAT` (used for the Monte Carlo calculations) and associate regions of the geometric mesh with materials. Each *Material* statement defines a different material. Materials are numbered in the order they appear in script. The maximum number of materials is 25. You can associated multiple regions with a single material. There are two forms of the command:

### **MATERIAL MatNo [Conductor, Insulator]**

The integer parameter *MatNo* corresponds to the number of a predefined Penelope material (1-279). For a list of available materials, see Sect. 17 or the `View material file` command in the *Gambet File* menu. The optional string parameter [`Conductor` or `Insulator`] overrides the default material state for the generation of Penelope cross sections (metallic elements in the range 1-99 are set as conductors, compound materials in the range 100-280 are set as insulators). If an `EFile` has been loaded, the `Conductor/Insulator` setting affects the default status for electric and magnetic field calculations inside the material. By default, the electric field is taken as zero inside the volume of conducting regions of the geometric mesh, independent of region definitions in the electric field file. The electric field has non-zero default values inside insulators and voids. Static magnetic fields are unaffected by the conductivity status. The material status for field calculation may be changed with the `ECalc` and `BCalc` commands.

## **MATERIAL**

### **NAME**

**COMPONENT [Chemical symbol or atomic number] [Fraction]**

### **DENSITY**

### **INSULATOR**

## **END**

A *Material* command with no parameters signals the beginning of a material section. You can define new materials with this construct. The section must terminate with the **End** command. The following commands may appear in any order within the material section:

### **NAME MatName**

Enter a name for the material (1-80 characters). The name should not contain any of the standard delimiters. Examples of valid names: `Carbon_Dioxide`, `High_Density_Lucite`, ...

### **COMPONENT [Chemical symbol or atomic number] [Fraction]**

A material section may contain up to 20 **Component** commands. Each command defines an elemental component of the material. The second parameter (string or integer) is the chemical symbol or the atomic number (*i.e.* Ga or 31, C or 12). The third parameter (real) is the number of atoms per molecule or stoichiometric fraction. Fractional values are permitted for alloys and mixtures. For example, the following lines describe acetic acid ( $\text{CH}_3\text{CO}_2\text{H}$ ):

Component C 2.0

Component H 4.0

Component O 2.0

### **DENSITY**

Material density in  $\text{gm}/\text{cm}^3$ .

### **INSULATOR, CONDUCTOR**

Conductivity state of the material

## **REGION RegNo MatNo**

This command associates a region of the geometric mesh with a defined material. The integer parameter **RegNo** is the region number and **MatNo** is the material number. For example, to specify that region 6 consists of the third material to appear in the **Composition** section, use the command:

```
REGION(6) = 3
```

**Gambet** issues an error message if any regions remain unassigned or if *MatNo* exceeds the number of defined materials.

### **REGION RegNo VOID**

Assign the *void* property to region *RegNo*. Particles move through voids without interaction. Electric and magnetic fields may act on electrons or positrons in voids. You can speed calculations by representing a region with infrequent interactions (*i.e.*, a low-density gas) as a void.

### **REGION RegNo EXCLUDE**

Assign the *exclusion* property to region *RegNo*. Particles that enter an exclusion region are immediately stopped and their properties at the region boundary are recorded in the escape file. In other words, they are treated in the same way as particles that leave the solution volume. This command is useful for defining a non-planar exit boundary in a three-dimensional solution. It may also be applied to find the distribution of particles entering a region without tracking interactions inside the region.

### **ECALC RegNo [ON, OFF]**

Over-ride default settings to specify whether electric fields are zero [*Off*] or non-zero [*On*] inside a region. This command has an effect only when an **EFile** has been loaded.

### **BCALC RegNo [ON, OFF]**

Over-ride default settings to specify whether magnetic fields are zero [*Off*] or non-zero [*On*] inside a region. This command affects all sources of magnetic field (**BFILE**, **BTABLE**, **BUNI**).

### **FULLINFO**

If this command is issued, **Gambet** records full information reported by the Penelope sub-routines during the creation of the file **GAMBET.MAT**. This information is useful only to users familiar with the intricacies of Penelope.

---

## 8 Source commands

COMMAND SUMMARY
<pre>SLIST (Data lines) END SFILE SourceFilePrefix PLIST (Data lines) END PFILE PartFilePrefix NPMULT NPerPrimary TPULSE TPulse SHIFT XPShift YPShift ZPShift ROTATE RotPX RotPY RotPZ [RotOrderP]</pre>

**Source** commands define the set of primary particles that initiate showers of electrons, photons and positrons. Here, we use the term *shower* to refer to the set of interactions and secondary particles that occur for each instance of an injected primary particle. For example, if there is a single primary particle and the parameter  $N_{pmult} = 500$ , then **GamBet** initiates 500 showers by injecting the primary particle 500 times with a statistical weighting factor of 1/500.

There are four ways you can specify primary-particle information:

1. Lists of particles that you prepare with an editor, spreadsheet or your own software.
2. Source files with large numbers of particles generated by the **GenDist** utility from script commands.
3. The particle escape file from a previous **GamBet** run.
4. Output files of electron or positron parameters from the programs **Trak** or **OmniTrak**.

The maximum number of unique primary particles is 1,000,000. You can initiate multiple showers for each primary particle with the *NPMult* command.

It is important to clarify the definition of a primary particle:

- Each data line in the **SList**, **SFile**, **PList** or **PFile** commands defines a primary particle.
- The total particle flux into the solution volume is the sum of the fluxes of primary particles. For example, if the list contains 10 electrons that each carry 1 mA of current, the injected current is 10 mA.

- Primary particles may be split into multiple showers using the `NPMult` command for improved statistics. If  $N_{pmult} > 1$ , then each primary particle is injected  $NPMult$  times with a weight of  $1.0/N_{pmult}$ . For example, if the list contains one electron that carries current  $2.0 \mu\text{A}$  and  $N_{pmult} = 20$ , then 20 shower calculations are initiated by identical electrons that each carry  $0.1 \mu\text{A}$ .

## 8.1 Particle input from source files

### SLIST

(Data lines)

### END

The purpose of the *SList* structure is to read input particle data from source data lines in the **Gambet** script. Each primary particle in the *SList* structure is defined by a data line. A line may contain 8 or 9 quantities:

```
Type  Kinetic  x   y   z   ux   uy   uz  [Current/flux]
      energy
```

Line entries may be separated by any valid delimiters. The quantities in the line have the following meanings:

#### Type

A string quantity that gives the type of particle. The following entries are recognized:

Electron: "1", "E" or "E-"

Photon: "2" or "P"

Positron: "3" or "E+"

#### Kinetic energy

The particle initial kinetic energy in electron volts ( $1.0\text{eV} = 1.6021 \times 10^{-19}$  joules).

#### x, y, z

The initial particle position in meters or units set by `DUnit`. The initial position must be inside the boundaries of the geometric mesh. If you want to start particle at a boundary, pick a position slightly inside the solution volume.

#### ux, uy, uz

A vector parallel to the initial particle direction. **Gambet** converts the quantities to a unit vector.

#### Current/flux

An optional parameter to represent the current carried by primary electrons and positrons or the flux of photons. For cylindrical or 3D geometric meshes, enter the current in A or flux in photons/s. For a planar geometric mesh, enter the line current density  $J$  in A/m or the line flux in photons/s/m. Note: if a primary particle carries current  $I$  and **Gambet** creates  $NPMult$  showers per primary, the input current assigned to each shower is  $I/N_{pmult}$ .

As an example, the following line designates a 1.7 MeV electron beam moving in the  $z$  direction at position (1.0, 2.0, 0.0) carrying a current of 1.0  $\mu\text{A}$ :

```
E- 1.7E6 1.00 2.00 0.00 0.00 0.00 1.00 1.0E-6
```

### SFILE SPrefix

This command directs **GamBet** to read particle data lines from an external file with a name of the form **SPrefix.SRC**. The file prefix contains from 1 to 32 characters. The file must be available in the working directory. A particle file may consist of blank lines, comments and from 1 to 20000 data lines. Data lines have the same format as those of the **SList** command. The file must terminate with the **End** or **EndFile** command.

## 8.2 Particle input from Trak and OmniTrak

Standard particle list files (**PName.PRT**) created by **Trak** and **OmniTrak** contain a data line for each particle. The list may contain blank and comment lines and must terminate with the **End** command. Comment lines are marked by the asterisk (\*) symbol. Particle information lines may be inserted directly into the **GamBet** script with the **PList** construct or they can be read from an independent file with the **PFile** command. For both the **PList** and **PFile** commands, **GamBet** expects a series of text lines consisting of nine or ten real numbers.

### PLIST

(Data lines)

#### END

The **PList** construct reads particle parameters from data lines in **Trak** or **OmniTrak** format in the **GamBet** script. Data lines have the following components:

```
Rest  Charge  Kinetic  x  y  z  ux  uy  uz  [Current]
mass          energy
```

Line entries may be separated by any valid delimiter. The quantities in the line have the following meanings:

#### Rest mass

The particle rest mass in AMU (atomic mass units), where  $1.0 \text{ AMU} = 1.65979 \times 10^{27} \text{ kg}$ . The rest mass of electron and positrons is  $5.488 \times 10^{-4} \text{ AMU}$ . If 0.00 appears in this position, **GamBet** will insert the rest mass of the electron/positron. The program issues an error message if the list contains particles other than electrons and positrons.

#### Charge

The particle charge in units of elementary charge,  $1.60210 \times 10^{-19} \text{ coulombs}$ . **GamBet** issues an error message if this entry does not equal -1.0 (electrons) or +1.0 (positrons).

#### Kinetic energy

The initial particle kinetic energy in electron volts ( $1.0 \text{ eV} = 1.6021 \times 10^{-19} \text{ joules}$ ).

**x, y, z**

The initial particle position in meters or units set by `DUnit`. The initial position must be inside the boundaries of the geometric mesh.

**ux, uy, uz**

A vector parallel to the initial particle direction. **GamBet** normalizes the values to create a unit vector.

**Current**

An optional parameter to represent the current associated with the primary particle. Specify the current in amperes for cylindrical or 3D geometric meshes. For a planar geometric mesh, the current represents the line current density  $J$  of a sheet particle in A/m.

**PFILE FPrefix**

This command directs **GamBet** to read particle parameter lines from an external file with a name of the form `FPrefix.PRT`. The file prefix contains from 1 to 46 characters. The file must be available in the working directory. A particle file may consist of blank lines, comments and from 1 to 1,000,000 data lines. Data lines have the same format as those of the `PList` command. The file must terminate with the `EndFile` command.

## 8.3 Number of showers and source properties

**NPMULT NPMult**

The integer parameter  $NPMult$  is the number of showers to initiate per specified primary particle. For example you could represent a uniform current-density electron beam with 200 primary particles distributed in radius and choose  $N_{pmult} = 500$  to achieve good statistics. In this case, the total number of showers is  $200 \times 500 = 100,000$ . For a specified primary particle, each particle that initiates one of the 500 showers starts in the same position but follows a statistically-independent history. The initiating particle has a weight of  $1.0/500$  for the calculation of dose and flux.

**TPULSE TPulse**

This command affects how dose is recorded in the output files. Enter the value of the beam pulselength in seconds. For example, suppose the sum of primary particle currents is 2.5 A. In the absence of the `TPulse` command, **GamBet** calculates the dose rate that would result from a continuous 2.5 A beam. If the command  $T_{pulse} = 2.0 \times 10^{-3}$  appears, the code calculates the accumulated dose resulting from an injected charge of  $5.0 \times 10^{-3}$  coulomb.

**ROTATE PRotX, PRotY, PRotZ [RotOrder]**

Rotate the particle direction vectors relative to the geometry mesh. Enter the values of angles in degrees. The optional string parameter *RotOrder* specifies the order of rotations (*i.e.*, 'ZX', 'ZYX'). Only the operations specified in `RotOrder` are performed. The default is *RotOrder* = 'XYZ'. Note that rotations are applied before position shifts.

**SHIFT XPSHift YPSHift ZPSHift**

Shift the initial positions of particles relative to the geometry mesh. Enter values in units set by `DUnit`. Note that rotations are applied before shifts.

---

## 9 Process commands

### COMMAND SUMMARY

#### Penelope control

```
EABS [Electron,Positron,Photon] 10000.00 [MatNo]
C1 0.1 [MatNo]
C2 0.1 [MatNo]
WCC 100.0 [MatNo]
WCR 100.0 [MatNo]
RSEED ISeed1 ISeed2
EMAX EMax
FORCE [BREMS,PAIR,...] FFactor [NReg]
ENHANCE NReg NSplit
REDUCE NReg NKill
```

#### Run control

```
DSMAX DsMax
DSMAX NReg DsMax
TIME 600
STEPMAX 20000
PLOTON [NPInterval]
      [ElecP,PhotP,PosiP,ElecS,PhotS,PosiS]
NSEARCHWIDTH G [2]
STEPLIMIT
MCA NBins Reg01 Reg02 ... Reg10
NOESCAPE
ESCAPEFILTER Condition01 Condition02
```

### 9.1 Penelope control

The first set of commands discussed in this section control operation of the Penelope package. To obtain the detailed Penelope manual, click on the **Penelope manual** command in the **Help** menu of **GamBet**. Although a detailed knowledge of the underlying physics of Penelope is not required to run **GamBet**, you should be aware that the parameters may affect run accuracy and speed. The following material is adapted from the Penelope manual.

### **EABS [Electron,Positron,Photon] 10000.00 [MatNo]**

A particle in material *MatNo* is assumed to be absorbed when its energy drops below a value *EAbs* (in eV). With this command, you can set independent absorption energies for electrons, photons and positrons. Because the interaction database is limited to energies above 50 eV, absorption energies *EAbs* must be larger than this value. If a value for *MatNo* is not given, the absorption energy is applied in all materials. Note that positrons undergo annihilation and emit two photons when they are absorbed. (Default value: 10 keV for electrons, photons and positrons).

The absorption energy values, particularly those of electrons and photons, can significantly affect the solution run time. Generally, there is no reason to track low-energy electrons with range small compared to the system size if you are not interested in details of the low-energy particle spectra. Furthermore, small values of *EAbs* for electrons and positrons can result in poor solution accuracy in the presence of magnetic fields (see the discussion under the *DsMax* command).

### **C1 0.1 [MatNo]**

The parameter *C1* marks the division between discrete and continuous calculations for electron or positron scattering. The quantity equals the average angular deflection ( $1 - \cos(\theta)$ ) produced by multiple elastic scattering along a path length equal to the mean free path between hard elastic events. The default value is  $C_1 = 0.05$ , and the maximum allowed value is 0.2. If *MatNo* is not specified, the value is applied in all materials of the solution volume.

### **C2 0.1 [MatNo]**

The parameter *C2* marks the division between discrete and continuous calculations of electron or positron energy loss. The quantity equals the maximum average fractional energy loss between consecutive hard elastic events. The default value is  $C_2 = 0.05$ , and the maximum allowed value is 0.2. If *MatNo* is not specified, the value is applied in all materials of the solution volume.

### **WCC 100.0 [MatNo]**

The parameter *Wcc* is the cutoff energy loss (in eV) for hard inelastic electron collisions. If *MatNo* is not specified, the value is applied in all materials of the solution volume. The default is  $W_{cc} = 1000$  eV.

### **WCR 100.0 [MatNo]**

The parameter *Wcr* is the cutoff energy loss (in eV) for hard bremsstrahlung emission. If *MatNo* is not specified, the value is applied in all materials of the solution volume. The default is  $W_{cr} = 1000$  eV.

The control parameters determine the accuracy and speed of the simulation. For highest accuracy at low energy the quantities *C1* and *C2* should have small values (0.01). With larger values the simulation gets faster at the expense of some loss in accuracy. The cutoff

energies  $W_{cc}$  and  $W_{cr}$  mainly influence the simulated energy distributions. The simulation speeds up by using larger cutoff energies, but if these are too large the simulated low-energy distributions may be somewhat distorted. In practice, simulated energy distributions are found to be insensitive to the values of  $W_{cc}$  and  $W_{cr}$  when they are less than the bin width used to tally the energy histograms. Thus, the desired energy resolution determines the maximum allowed cutoff energies. The reliability of the whole simulation rests on a single condition: the number of steps per primary track must be 'statistically sufficient' (*i.e.* larger than  $\sim 10$ ).

### **RSEED ISeed1 ISeed2**

Set a random seed for the simulation by supplying two integer values. You can use this command to ensure that a simulation generates identical results each time it is run. The default is to choose seed values set by the current clock time to generate statistically independent results in each run.

### **EMAX EMax**

Sets a maximum energy for the cross-sections generated in `GAMBET.MAT`. In a solution with electric fields, be sure to allow some slack to account for acceleration and/or numerical inaccuracies in the orbit simulation. If the `EMax` command does not appear, **Gambet** estimates a value from the energies of particles defined in the *Source* section. For an incident positron with kinetic energy  $T_e$ , the maximum energy is taken as  $E_{max} = 1.21 \times (T_e + 5.11 \times 10^5)$  eV to account for the kinetic energy of annihilation photons.

## **9.2 Variance reduction control**

The next set of commands addresses processes often gathered under the term *variance reduction*. Here the intent is to improve statistics for a given run time by devoting resources to critical interactions of interest while de-emphasizing reactions that have little effect on the results. You must perform sufficient checks to ensure that results derived under the following commands are physically correct for your application (*i.e.*, making test runs with and without the application of variance reduction).

### **FORCE [BREMS,PAIR] Factor [NReg]**

Define a forcing factor to increase the number of particles generated as the result of low-probability events. **Gambet** automatically adjusts particle weights and interaction probabilities to maintain unbiased statistics. The second keyword gives the interaction type. Two options are available: *Brems* (generation of bremsstrahlung radiation by energetic electrons and photons) and *Pair* (creation of electron/positron pairs by energetic photons). The third parameter is the forcing factor for the interaction (a real number larger than 1.0). The final integer parameter *NReg* is the number of the region in the geometric mesh where forcing should be applied. If *NReg* does not appear, forcing applies in all regions. Note that large forcing factors ( $> 100$ ) may skew the statistics and also exhaust the available storage in the secondary-particle stack. **Gambet** includes a warning in the listing file if stack overflow errors occurred during a run.

### **ENHANCE NReg NSplit [ElecP, PhotP, PosiP, ElecS, PhotS, PosiS]**

You can use this command to improve statistics in particular regions of the solution volume. Any particle that enters region *NReg* is split into *NSplit* particles with weights adjusted by a factor  $1.0/N_{split}$  to preserve unbiased statistics. The integer quantity *NSplit* (the enhancement factor) must be greater than 1. This command is most effective if it is applied to a remote region that receives a small fraction of the particle flux created in the solution. The script may contain multiple **Enhance** commands. If you include from 1-6 optional string parameters, **GamBet** applies enhancement only to the specified classes of particles: *ElecP* (primary electrons), *PhotP* (primary photons), *PosiP* (primary positrons), *ElecS* (secondary electrons), *PhotS* (secondary photons), *PosiS* (secondary positrons).

### **REDUCE NReg NKill [ElecP, PhotP, PosiP, ElecS, PhotS, PosiS]**

This command reduces the code resources devoted to particle tracking in region *NReg* by applying a Russian-roulette technique. A particle that enters the region is killed with probability  $P_{kill} = 1.0 - 1.0/N_{kill}$ . If it survives, its weight is increased by a factor  $1.0/(1.0 - P_{kill})$ . The integer quantity *NKill* (the reduction factor) must be greater than 1. This command is useful if the solution volume contains a large, remote region that makes only a small contribution to interactions in the solution volume. A script may contain multiple **Reduce** commands. If you include from 1-6 optional string parameters, **GamBet** applies reduction only to the specified classes of particles: *ElecP* (primary electrons), *PhotP* (primary photons), *PosiP* (primary positrons), *ElecS* (secondary electrons), *PhotS* (secondary photons), *PosiS* (secondary positrons).

## **9.3 Simulation control**

The remaining commands control general features of the simulation.

### **DSMAX [NReg] DsMax**

**GamBet** tracks electrons and positrons that interact with materials and move in electric and magnetic fields by dividing the trajectory into a number of small steps of length *Ds*. This command may be used to set a global value or individual region values for the maximum step size. The default value for all regions is the diagonal length of the solution volume divided by 100.0. Enter the value in units set by *DUnit*. In the default program mode, this command has no effect on photon orbits which generally move in large steps between discrete interactions.

Use the following guidelines to pick values of *DsMax*.

1. The Penelope routines usually assign small values of *Ds* for electrons moving in materials. In this case, the value of *DsMax* has little effect on calculations with no applied fields. For high-energy electrons incident on thin foils, set *DsMax* less than or equal to 1/10 of the foil thickness.
2. The value of *Ds* is critical in simulations with applied magnetic fields. It must be small compared to the radius of gyration of the minimum-energy electron or positron (usually determined by *EAbs*). A larger value of *Ds* gives inaccurate or unstable orbits with poor energy conservation. Because small values of *Ds* may increase the run time, pick *EAbs* as large as possible in calculations with strong magnetic fields.

3. Penelope does not determine values of  $Ds$  in VOID regions. You must set a value for  $DsMax$  that ensures accuracy of the numerical orbit integrations if there are applied fields.
4. Check the energy conservation statistics computed by **GamBet** to determine whether  $Ds$  is small enough for accurate orbit calculations in applied magnetic fields. With inaccurate orbit integrations, the output energy may be higher than the input energy (global energy conservation). Section 10 discusses entries for energy conservation in the listing files.
5. By default, the maximum step size also applies to photons except when the **NoStepLimit** command appears in the script. Eliminating the step limit for photons saves time, but could cause invalid results when the solution volume contains a spatial region that 1) is surrounded by another region, and 2) has width comparable to or less than the photon mean-free path. Without a limit on step-size, photons may jump over the small region with sensing a boundary crossing, giving errors in the assigned dose.

## STEPMAX

Depending on the settings of  $EAbs$ ,  $C1$ ,  $C2$ ,  $Wcc$  and  $Wcr$ , the number of steps in an electron or positron history may be quite large. The parameter  $StepMax$  sets a maximum value to prevent infinite run times. The default is  $StepMax = 500,000$ . If a particle stops early, either increase  $StepMax$  or change the values of the Penelope control parameters. The listing file contains a warning message if any particle trajectory exceed  $StepMax$ .

## TIME 600

Use this command to set a maximum run time for the calculation. Enter the value in seconds. You can use this command to set a maximum time instead of a maximum number of showers. **GamBet** records complete information for showers calculated within the allowed time and completes the present shower before terminating. In a parallel run, the primary and all child processes will stop at the time limit.

## NSEARCHWIDTH [G, E, B] NSearchWidth

**GamBet** uses high-speed, high-reliability methods to search meshes for particle positions. The default search width of 3 elements should be sufficient for all circumstances. Increase  $NSearchWidth$  if the program reports a mesh search error. For simple meshes, the program may run faster with a lower value. You can set individual values for searches on the geometric, electrical and magnetic meshes.

## PLOTON [NPShower] [ElecP, PhotP, PosiP, ElecS, PhotS, PosiS]

In response to this command, **GamBet** records selected trajectories of primary and secondary particles. The trajectories may be displayed in **GBView2** and **GBView3**. The integer quantity  $NPShower$  is the number of showers to record for each primary. The default is  $N_{pshower} = 1$ . If you include from 1-6 optional string parameters, **GamBet** records only trajectories for certain types of particles:  $ElecP$  (primary electrons),  $PhotP$  (primary photons),  $PosiP$  (primary positrons),  $ElecS$  (secondary electrons),  $PhotS$  (secondary photons),  $PosiS$  (secondary

positrons). Use **this command with caution**: a run with a large number of primaries or with  $N_{pmult} \gg 1$  can generate a huge plot file. Plotting is deactivated in parallel runs.

### **PARALLEL NProc [TStart TEnd]**

Run multiple instances of **GamBet** on a multi-processor computer. The integer parameter *NProc* equals the total of the primary and child processes, usually equal to the maximum number allowed on the computer (*e.g.*, for a Core i7 machine, the maximum value is 8). The integer *TStart* is the maximum time (in seconds) to wait for the start of each child process, and *TEnd* is the maximum time to wait for the conclusion. Section 11 gives a detailed description of the parallel procedure and factors that influence the choice of parameters.

### **NOSTEPLIMIT**

The maximum step size applies to photons except when the **NoStepLimit** command appears. Eliminating the step limit for photons may save time, but large photon steps can lead to invalid results when the solution volume contains small regions that are immersed in another region. In this case, **GamBet** may fail to sense a boundary transition for steps that cross the small region but have starting and ending points in the surrounding region.

### **ESCAPEOFF**

Suppress creation of the file of escaping particle parameters.

### **ESCAPEFILTER Condition01 Condition02 ...**

By default, the escape file contains all particles that leave the solution volume. Filters may be applied that limit particles recorded in the escape file by species, exit position and kinetic energy. This command is useful to reduce the size of escape files by limiting data to particles of interest, thereby reducing the load time in **GenDist**. The nature of the filter is determined by specified conditions that follow the command. A condition is short string that does not contain any of the standard delimiters. Conditions are separated by spaces. Any number of conditions may be included in the line. The following conditions determine the included particle types (primary or secondary): **ElecP**, **PhotP**, **PosiP**, **ElecS**, **PhotS** and **PosiS**. If none of the conditions appear, all particle types are included. If one or more conditions appears, only the specified types are included. For example, this command includes only secondary electrons and photons:

```
ESCAPEFILTER ElecS PhotS
```

The following expressions define pass conditions for particles according to their exit positions:  $X > 10.56$ ,  $Y < 1.0E-8$ ,  $Z > 15$ ,  $R > 30.5$ , .... The expression may not include spaces or other delimiters. The condition must begin with the character *X*, *Y*, *Z* or *R*. The quantity *R* is given by

$$R = \sqrt{X^2 + Y^2}. \quad (1)$$

The next character must be either  $>$  (greater than) or  $<$  (less than). The remainder of the condition is a real number in any valid format. Specify the coordinate in units given by the *DUnit* parameter. As an example, the command

```
ESCAPEFILTER X>0.0 Y>0.0
```

passes only particles that exit in the first quadrant of the  $x$ - $y$  plane. In this case, no filtering is applied in  $z$ . Finally, conditions of the form  $T<5.0E6$ ,  $T<1500$ ,.... pass particles according to their kinetic energy. Specify energy values in eV. As an example, consider a 500 keV photon beam passing through a slab of material with  $z_{max} = 60.0$  mm. To calculate the attenuation coefficient, we want to find the number of full-energy, primary photons that emerge from the exit face. The appropriate filter command is:

```
ESCAPEFILTER PhotP T>4.999E5 Z>59.999
```

### MCA NBins Reg01 Reg02 ... Reg10

The main use of the MCA (Multi-channel Analyzer) command is to characterize detectors. When the command is issued, **GamBet** opens a file `RunName.PLS` and records total deposited energy in selected regions for each shower. At the end of run, the program uses values in the file to create a histogram for each region showing the number of showers per range of deposited energy. The histograms are recorded in the listing file `RunName.GLS`. The integer parameter  $NBins$  is the number of energy intervals. The allowed range is  $N_{bins} = 2$  to 250. **GamBet** automatically sets minimum and maximum values for the energy range. The integer parameters  $Reg01$ ,  $Reg02$ ,... are the numbers of any regions in the geometry mesh with non-zero volume. You can enter a maximum of 10 regions. The MCA command functions only in the `SingleParticle` mode. Also, the command may not be used in a run with the `Parallel` command.

A histogram listing is shown below. A plot of the number of showers versus energy looks like the display of a multichannel analyzer monitoring a scintillation or solid state detector.

```
Deposited energy distribution for Region 1
  Energy(min):  0.0000E+00
  Energy(max):  2.1670E+06
    dEnergy:    2.1670E+04
    EDn         EUp         NEvent
    (ev)        (eV)
=====
  0.0000E+00  2.1670E+04      10996
  2.1670E+04  4.3340E+04         59
  4.3340E+04  6.5010E+04         62
  6.5010E+04  8.6680E+04         54
  ...
  2.0803E+06  2.1020E+06          2
  2.1020E+06  2.1237E+06          0
  2.1237E+06  2.1453E+06          4
  2.1453E+06  2.1670E+06      2287
```

---

## 10 Running GamBet: interactive and background modes

A basic **GamBet** run involves the following activities:

1. Create an input script (**RunName.MIN**) describing the system geometry. Run the **Mesh** (2D) or **MetaMesh** (3D) program to convert the geometric specifications to a conformal mesh file.
2. Create an input script (**RunName.GIN**) to control the program and to define the properties of material and sources. Run **GamBet** to generate output files that describe spatial variations of dose, particle orbits within the solution volume and parameters of particle that escape.
3. Analyze the output files using **GBView2**, **GBView3** or **GenDist**.

The section describes the second step, operation of **GamBet** in either the interactive or background modes.

The program **gambet.exe** can run interactively in a Window. In this mode you can carry out several solutions in a session or leave the program while you work in other Windows applications. The most convenient way to run **GamBet**, **GBView2** or **GBView3** is from the **GBLaunch** program launcher. You can also run the executable programs individually from Windows Explorer or add shortcuts to the desktop or the Windows Start Menu.

**GamBet** has three popup menus: **File**, **Run** and **Help**. The following commands appear in the **File** menu.

### CREATE INPUT FILE

This command calls up a full-featured internal Windows editor to create an input script. Supply a run prefix (1-46 characters). The allowed commands appear as a set of comment lines. Save or abandon the file to return to the solution program.

### EDIT INPUT FILE

### EDIT LISTING FILE

### EDIT FILE

These commands call the editor to inspect or to modify **GamBet** input and output files. Choosing a file from an alternate directory does not change the working directory. The **Edit input file** command shows a list of all files with names of the form **FPREFIX.GIN**. The **Edit listing file** command displays files with names **FPREFIX.GLS**.

The **Run** menu contains four commands:

## **START RUN**

Pick an input file (`FPREFIX.GIN`) to start a solution. The working directory is changed if you pick a file from an alternate directory. The run begins if the geometric mesh file and all required data files are present. The small text window displays information on the run status.

## **PAUSE RUN**

The intensive calculations of the Monte Carlo simulation demand the full resources of your computer, causing other tasks to run slowly. If you need to perform critical work, you can pause the solution program and reactivate it later without loss of data.

## **STOP RUN**

This command terminates the program and saves information after completing the current shower.

The *Help* menu has the following entries:

## **GAMBET REFERENCE MANUAL**

Open this manual in the default PDF viewer.

## **PENELOPE MANUAL**

Open the users' manual for Penelope 2006 in the default PDF viewer.

## **PHYSICAL CONSTANTS**

Display a useful list of physical constants in the default PDF viewer.

## **NIST ELECTRON DATA**

Go to the NIST EStar site in the default browser to find electron cross sections and ranges.

## **NIST PHOTON DATA**

Go to the NIST XCom site in the default browser to find photon cross sections.

## **FIELD PRECISION WEBSITE**

Go to the Field Precision home page in the default browser to check on other products or to contact us.

To run **GamBet** in the background, go to the command prompt from Windows (*i.e.*, run `CMD`) and log to the data directory that contains the control script (`FPREFIX.GIN`) and data input files. For example, suppose `COPPER_FOIL.MOU` and `COPPER_FOIL.GIN` are stored in `SIMULATIONS\WORKING` and that the program `gambet.exe` is in the directory `\GAMBET`. From `\SIMULATIONS\WORKING` type

```
C:\GAMBET\GAMBET COPPER_FOIL <Enter>
```

The program runs silently, writing detailed information in the listing file `COPPER_FOIL.GLS`. If the solution is successful, the program may write `COPPER_FOIL.G2D` or other files to the data directory. During lengthy runs you can perform other tasks in Windows.

The main function of the command mode is autonomous operation under batch file control. This feature is useful if you want to make an extended series of solutions overnight or in the background. As an example, suppose you have prepared the input files `SWT01.MIN`, . . . , `SWT08.MIN` and `SWT01.GIN`, . . . , `SWT08.GIN` in the directory `\SIMULATIONS\WORKING`. Furthermore, you have written the following batch file `SWRUN.BAT` in the working directory using a text editor.

```
@ECHO OFF
ECHO Main switch data run
START /WAIT C:\GAMBET\MESH.EXE SWT01
START /WAIT C:\GAMBET\GAMBET.EXE SWT01
START /WAIT C:\GAMBET\MESH.EXE SWT02
START /WAIT C:\GAMBET\GAMBET.EXE SWT02
. . .
START /WAIT C:\GAMBET\MESH.EXE SWT08
START /WAIT C:\GAMBET\GAMBET.EXE SWT08
```

Note the use of `/WAIT` option. The mesh data must be available before the **GamBet** calculations. Type

```
SWRUN<Enter>
```

to generate all solutions without the need for further user input. Sometimes you can make efficient use of a multicore computer by eliminating the `wait /WAIT` directive and launching multiple instances of the program. In this case, you must ensure that individual processes do not try to access the same input/output files.

---

## 11 GamBet: parallel operation

**GamBet** run times may be significantly reduced on modern personal computers with parallel techniques. Field Precision programs are designed for Windows-based machines with multiple processors or cores – a *shared-memory architecture*. Accordingly, we have integrated **OpenMP** in our field solution programs such as **HiPhi** and **Magnum**. For these codes, the matrix-inversion process that occupies the bulk of the run time is relative easy to parallelize. Threads work on separated parts of the matrix so there is no possibility of memory conflicts.

Although at first glance a Monte Carlo code would seem to be a good candidate for parallel operation, there are potential issues in **GamBet**:

- Primary and secondary particles update deposited dose on a single conformal mesh. With complex trajectories, it is not possible to ensure that particles from parallel showers do not occupy the same cell.
- The **Penelope** library was not written for parallel operation and there are potential conflicts in shared resources like the secondary stack.

For these reasons, we decided to take the *distributed memory* approach with **GamBet**. In this case, multiple instances of the program with independent storage run in individual threads controlled by the operating system. The resulting data files are combined at the end to a single representation with improved statistics. This approach is practical for current computers with large memories. Furthermore, it is easily extended to cluster computers.

The **GamBet** procedure is as follows:

1. In a parallel run, the executable `gambet.exe` (the *primary* process) may run either interactively or in the background. In turn, it launches multiple instances of the background program `gambetmp.exe` (*child* processes). All processes utilize the same input files (control script, geometry and field meshes,...).
2. A new command in the **Process** section of the input script signals that the primary process should start a parallel run:

```
PARALLEL NProc [TStart TEnd]
```

The integer parameter *NProc* specifies the number of threads. For a Core i7 processor, the maximum value is  $NProc = 8$ . Guidelines for picking *NProc* are discussed below. The parameter *TStart* is the number of seconds to wait for a signal from each child process before abandoning the launch. The parameter *TEnd* is the maximum number of seconds to wait for each child process to finish before combining the output data files.

3. If the primary process detects the **PARALLEL** command, it creates a common **Penelope** material input file `GAMBET.MAT` and then sequentially initiates ( $NProc-1$ ) instances of the program `gambetmp.exe`. After launch, a child process reads the shared input files. The

process sends a start signal to confirm that the input files have been closed, a precaution against access conflict. The signal is the creation of a blank file. If the control script has the name `FPREFIX.GIN`, the start signal files have names of the form `FPREFIX01.START`, `FPREFIX02.START`,.... When the primary process detects a start signal, it notes a successful launch and proceeds to the next child process. The primary process seeks the start signal for each child process for a maximum of  $TStart$  seconds.

4. When all child processes are running, the primary process proceeds with its own calculation, issuing standard screen messages for interactive runs.
5. When the calculation is complete, the primary process checks that all valid child processes have sent an end signal (*i.e.*, created a file with a name of the form `FPREFIX01.END`). The program waits a maximum of  $TEnd$  seconds for each process to finish.
6. The data of the primary process and successful child processes are combined. Start and end signals and unnecessary data files are deleted.

Although the sequence of a parallel run is involved, all operations are organized by the primary process. The user need only add the `PARALLEL` command. The primary process creates a text report, `MPSTATUS.DAT`. Check this file if there are any problems.

Data files are combined according to the following rules:

- Each process creates an escape file listing parameters of particles that escape from the solution volume (`FPREFIXESC.SRC`). Data lines from the child files are appended to the primary escape file. If particles have an associated current or flux, the values are divided by  $NProc$ . When complete, the child escape files are deleted. To interpret the number of entries in the combined escape file, note that the total number of showers per primary particle is  $NPMult \times NProc$ .
- Dose files describe the spatial distribution of deposited energy in the solution volume. They have names of the form `FPREFIX.G2D` (two-dimensional) and `FPREFIX.G3D` (three-dimensional). Information from dose files created by the primary and child processes is accumulated, averaged and then written to the primary dose file. The child dose files are deleted.
- The listing files from the primary and child processes contain run statistics, information on deposited energy and values for particle and energy flux between regions. These quantities are accumulated, averaged and then written to the primary listing file (`FPREFIX.GLS`). Child listing files are deleted.
- **GamBet** can record sample electron, photon and positron trajectories in a file with a name of the form `FPREFIX.GPL`. The purposes of trajectory files are to check run physics and to create presentations. They are generally not generated in production Monte Carlo runs. A run with millions of trajectories, each with thousands of orbit vectors, could result in unmanageably large files. Therefore, trajectory plots are automatically deactivated in parallel runs. To create plots, use a serial run and limit the number of trajectories per primary particle.

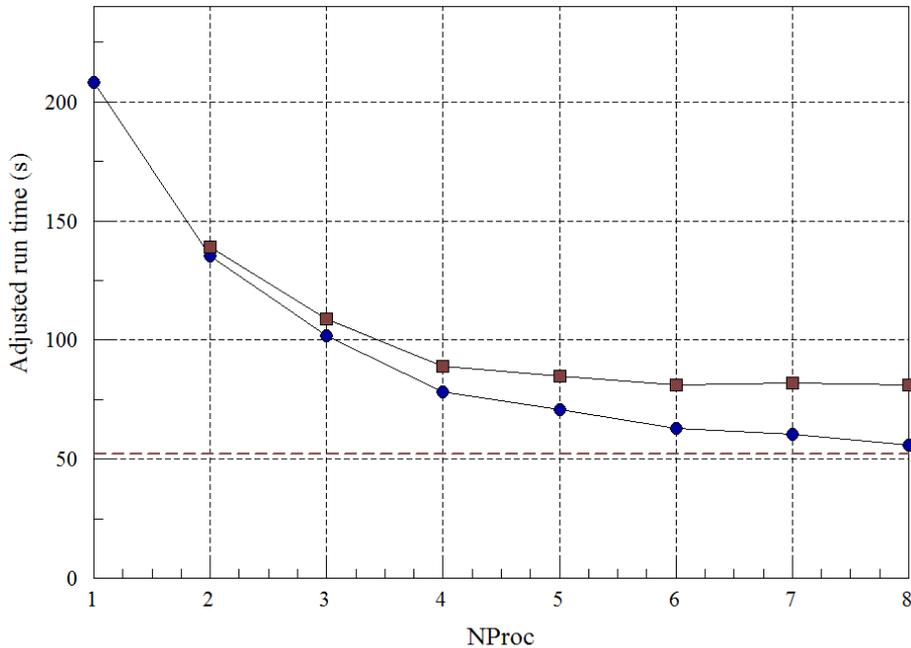


Figure 8: Benchmark test: run time as function of the number of processes. Red: Total run time. Blue: Time with input/output operations subtracted.

Next, we shall consider how to pick parameters in the `PARALLEL` command and how much performance improvement to expect. The maximum value of  $NProc$  is 99. The parameter should not be set higher than the maximum number of independent threads available on the computer. First consider the value of  $TStart$ . Each child process must read input files, and two processes cannot access a file at the same time. Therefore, the primary process must wait for a signal that the child process has read and closed all files. This normally requires only a few seconds, so the default  $TStart = 10$  seconds is sufficient. If the simulation involves a complex three-dimensional mesh or a large input `SRC` file, you may need to increase  $TStart$ . With regard to run completion, the child processes should be finished before the primary process if all threads have the same run time. The parameter  $TEnd$  is included for any operating-system vagaries. If the primary process does not detect the end signal from a child process, it will wait  $TEnd$  seconds before abandoning the attempt to combine files. The default is  $TEnd = 10$  seconds. Again, increase  $TEnd$  if there are problems reported in `MPSTATUS.DAT`.

To illustrate the effect of the  $NProc$  parameter, consider a benchmark example of an electron beam incident on an aluminum target. The total number of showers from all processes was fixed at 2,000,000. The run was performed on a Core i7 machine running at 2.93 GHz. The total run time as a function of  $NProc$  is shown as the red squares in Fig. 8. The serial run time was 208 seconds. Note that the total run time did not decrease above  $NProc = 6$ , saturating at a reduction factor of 1.72. The time required to read input files and to combine output files negated the speed increase from additional threads. A measurement showed that input/output operations occupied about 3.6 seconds/process. Subtracting this time gives the actual computation time, shown as the blue circles in Fig. 8. At  $NProc = 8$ , the time reduction approached the theoretical value of 4.0. The implication is that the full benefits of parallel operation can be achieved when the serial run time exceeds about 5 minutes.

Some notes on parallel operation:

- The programs `gambet.exe` and `gambetmp.exe` have been modified from Version 2.0 to ensure that there is a unique random number seed for the primary process and each child process. Otherwise, each process would generate identical information that would be summed and then divided by *NProc*. Without a unique seed, there would be no point to a parallel run.
- To minimize problems, always make sure that a run operates correctly in the serial mode before attempting a large parallel production run. For testing, remove the `PARALLEL` command and use a reduced value of *NpMult*.
- After launch, child processes run independently of the primary process. Therefore, the `Stop run` command in an interactive run will only stop the primary process. You must use the Windows task manager to terminate child processes.

---

## 12 Input/output files

During a run **GamBet** may generate four types of output data file

### LISTING FILE

In the default mode, simulations generate a listing file in text format with a name of the form `RunName.GLS`. The listing file contains extensive information about the run setup and also calculated statistics. The following section discusses information contained in the listing file.

### DOSE FILE

All simulations generate a dose file. Depending on the symmetry of the input *GFile*, the dose file has a name of the form `RunName.G2D` or `RunName.G3D`. Files of type `RunName.G2D` are analyzed with **GBView2** and files of type `RunName.G3D` with **GBView3**. The dose file contains the following information: parameters of the geometric mesh, region identities of nodes and elements, node coordinates, total dose in elements and dose contributions from primary and secondary electrons, photons and positrons and region properties. The units of recorded dose depend on the simulation mode. The **TriComp** program **TDiff** can import files of the type `RunName.G2D` to perform thermal analyses. In this case, the recorded element quantities must be dose rates; therefore, the calculation must be performed in the continuous beam mode (Chap. 3). Here, the current or flux information should be specified for the primary particles and *TPulse* must be undefined. A temporal function may be specified in **TDiff** to modulate the energy deposition rate. Similarly, files of the type `RunName.G3D` may be exported to the **AMaze** program **HeatWave**.

### ESCAPE FILE

In the absence of the **ESCAPEOFF** command, a simulation generates an escape file. This file lists properties of particles that terminate by leaving the material boundaries of the solution region. The file has a name of the form `RunNameECS.SRC` and the same format as the source file described in Sect.8.1. Therefore, an escape file can be used as the input file for a subsequent **GamBet** run or analyzed with the **GenDist** program. An escape event occurs when either 1) the particle moves outside the bounds of the geometric mesh, 2) enters a element with *RegNo* = 0 or 3) enters a region with the *Exclude* property (Sect. 7). The following quantities are included in comment lines at the beginning of the escape file:

#### DUNIT

Conversion factor to convert particle coordinates to meters.

#### NPRIMARY

Number of primary particles in the input list of the **GamBet** run.

#### NShower

Number of showers created in the **GamBet** run, where  $N_{shower} = N_{pmult} \times N_{primary}$ .

Table 6: Interpretation of flux in the **GamBet** escape file

Particle type	Mesh symmetry	Current/flux quantity
Electron/positron	3D, 2D cylindrical	A
	2D planar	A/m
Photon	3D, 2D cylindrical	photon/s
	2D planar	photon/m/s

Each data line in the file represents a simulation particle and contains the following entries:

- Particle type (E,P or E+)
- Kinetic energy (eV)
- Coordinates  $x,y$  and  $z$  (units defined by  $DUnit$ )
- Direction unit vector components  $u_x, u_y$  and  $u_z$
- Current/flux

The interpretation of the current/flux quantity depends on the geometric mesh symmetry and particle type (see Table 6).

### TRAJECTORY PLOT FILE

A shower trajectory file is created when the `PlotOn` command appears in the script. The file has a name of the form `RunName.GPL`. The header contains the quantities  $DUnit$  and  $NShower$ . The file contains trajectories of all primary and secondary particles in one or more showers. Each particle orbit starts with the heading:

```

=====
ShowerNo: 667
PrimaryNo: 4
PartType: [Electron, Photon, Positron]
          x          y          z          NReg
-----

```

and contains a number of data lines of the form

```
x,y,z,NReg
```

where the coordinates are given in units specified by  $DUnit$ . The line

```
-1.0000E+00 -1.0000E+00 -1.0000E+00 0
```

signifies the end of an orbit. Trajectories may be superimposed on dose plots in **GBView2** and **GBView3**. The file is in text format. The content may be inspected with an editor or ported to your own analysis programs.

## DEPOSITED ENERGY DISTRIBUTION FILE

The format of the file `RunName.PLS` is shown below. Values for the first region are listed in column 1, the second region in column2 and so forth. As an alternative to the histogram in the listing file, you can use the data directly in your own statistical analysis routines. There are six lines in the heading. The quantity *NLines* is the number of data lines, equal to the number of showers (*NPMult*). The unit (eV/p) denotes electron-volts per primary particle. Each data line contains from 1 to 10 entries, one for each region specified in the `MCA` command. The values are the deposited energy in each region for each shower.

GamBet Pulse-height Distribution File (Field Precision)

RunName: NaIDetect2

NLines: 20000

Regions: 3 6

Units: eV/p

```
0.0000E+00 0.0000E+00
8.2165E+05 1.3454E+06
0.0000E+00 0.0000E+00
0.0000E+00 7.7453E+05
0.0000E+00 0.0000E+00
...
```

---

## 13 Contents of the listing file

All Field Precision programs generate listing files that report on the progress of the run. The **GamBet** file `RunName.GLS` records run parameters and contains important results of the calculations.

### 13.1 Source information

After analyzing the commands of the *Source* section, **GamBet** lists the properties of the primary particle distribution. If the particle lists contain current or flux information, the reported averages are weighted by these quantities.

```
Input distribution properties
  Number of primary particles:      1
  Shower calculations per primary: 50000
  Total number of shower calculations: 50000
  Input energy: 1.0000E+06 (eV/m)
```

```
Electron properties
  Total number: 1.00
  Average energy: 1.0000E+06 (eV)
  Average position
    X: 1.0000E-04 (cm)
    Y: 0.0000E+00 (cm)
    Z: 0.0000E+00 (cm)
  Average direction unit vector
    U: 1.0000
    V: 0.0000
    W: 0.0000
```

### 13.2 Penelope setup information

After reading commands of the *Process* section, **GamBet** lists quantities generated by Penelope while building material tables and region properties.

Penelope setup

Specified control parameters

(Random number seeds: 12345 54321)

MatNo	EAbs(Elec) (eV)	EAbs(Phot) (eV)	EAbs(Posi) (eV)	C1
-------	--------------------	--------------------	--------------------	----

1	1.0000E+05	1.0000E+05	1.0000E+05	1.0000E-01
---	------------	------------	------------	------------

C2	WCC (eV)	WCR (eV)
----	-------------	-------------

1.0000E-01	1.0000E+05	1.0000E+05
------------	------------	------------

Set up material tables

Energy range, EMin: 1.0000E+05 (eV) EMax: 1.0000E+06 (eV)

Processing Material No.: 1

Electron absorption energy: 1.0000E+05 (eV)

Photon absorption energy: 1.0000E+05 (eV)

Positron absorption energy: 1.0000E+05 (eV)

C1: 1.0000E-01, C2: 1.0000E-01 WCC: 1.0000E+05 (eV) WCR: 1.0000E+05 (eV)

Mass density: 1.9320E+01 (g/cm3)

Number of elements in the molecule: 1

Element: Au (Z=79), atoms/molecule = 1.00000000E+00

Mean excitation energy = 7.9000E+02 eV

Maximum step sizes for electrons and positrons

NReg	DsMax
1	5.0000E-02 (cm)

Integrated region masses

NReg	Mass (kg/m)
1	5.7959E-01

### 13.3 Primary particle statistics

When the Monte Carlo calculation is complete, **GamBet** lists several categories of statistic information. The primary particle list has the format shown below. Note that escape particles are those primaries that leave the solution volume while absorbed particles stop inside the solution volume. Units depend on the simulation mode.

Primary electrons

Escape fraction: 0.4417

Average energy of escape electrons: 7.2260E+05 (eV)

Average pathlength of escape electrons: 1.3666E-02 (cm)

```

Average pathlength of absorbed electrons:  3.8120E-02 (cm)
Number of secondary electrons per primary:  0.290
Average secondary electron energy:  1.8428E+05 (eV)
Average energy lost to secondary electrons per primary:  5.3519E+04 (eV)
Number of secondary photons per primary:  0.141
Average secondary photon energy:  2.5054E+05 (eV)
Average energy lost to photons per primary:  3.5326E+04 (eV)
Number of secondary positrons per primary:  0.000
Average energy lost to positrons per primary:  0.0000E+00 (eV)

```

## 13.4 Energy conservation

Energy conservation statistics compare the input energy to total deposited and escape energies. In the absence of an applied electric field, the quantity *Energy difference* should be small ( $\ll 1\%$ ). Anomalous energy differences may appear under two circumstances:

- With chosen Penelope parameters, the value of *StepMax* may be too small. In this case, particle histories are not followed to completion.
- The step size *Ds* is too large for accurate numerical integrations in the presence of electric or magnetic fields.

To compute energy conservation, **GamBet** compares the total input particle kinetic energy to the sum of the escape energy and the energy deposited in the medium. **NOTE:** The energy difference may not be a small number if particles can gain significant energy within the simulation volume from applied electric fields.

### Energy conservation

```

Input energy:  1.0000E+06 (eV/m)
Energy deposited by primaries:  6.0399E+05 (eV/m)
Energy deposited by secondaries:  6.0372E+04 (eV/m)
Primary escape energy:  3.1917E+05 (eV/m)
Secondary escape energy:  1.6206E+04 (eV/m)
Energy difference:  2.6031E-02%
Energy deposited by electrons:  6.5512E+05 (eV/m)
Electron escape energy:  3.2095E+05 (eV/m)
Energy deposited by photons:  9.1827E+03 (eV/m)
Photon escape energy:  1.4430E+04 (eV/m)
Energy deposited by positrons:  0.0000E+00 (eV/m)
Positron escape energy:  0.0000E+00 (eV/m)

```

## 13.5 Dose assignment statistics

As a check **GamBet** prints a sum of energy deposited on the finite-element mesh. The quantities should equal the sum of primary and secondary deposited energy.

### Energy/Power deposited on the mesh

```

Global energy
Primary electrons:  3.3241E+04 (eV)
Primary photons:  0.0000E+00 (eV)
Primary positrons:  0.0000E+00 (eV)

```

```

Secondary electrons:  2.6175E+03 (eV)
Secondary photons:   8.8804E-02 (eV)
Secondary positrons: 0.0000E+00 (eV)
All particles:       3.5858E+04 (eV)

```

```

Region energy (eV)
NReg  Prim elect  Prim phot  Prim posit  ...
=====
  1  3.3241E+04  0.0000E+00  0.0000E+00
      Sec elect  Sec phot  Sec posit  Total
=====
      2.6175E+03  8.8804E-02  0.0000E+00  3.5858E+04

```

### 13.6 Particle flux statistics

Finally, **GamBet** makes a list of particle flux between regions. In the example listed below the solution volume contains a single region (numbered 1). Region number 0 represents any exterior volume. Therefore, the quantities shown below represent electrons and photons escaping from the solution volume. **NOTE:** In the example the electron flux value of 0.4998 is higher than the primary escape fraction of 0.4417 because of the expulsion of secondary electrons. In a solution with multiple regions, **GamBet** lists results for all boundaries that have non-zero flux.

```

--- Region fluxes ---
NReg  NReg  Electron  Electron  Photon  Photon
From  To    flux     eng. flux  flux    eng. flux
=====
  1    0    4.499800E-01  3.209489E+05  4.750000E-02  1.443030E+04

```

---

## 14 GBView2

**GBView2** is an interactive graphical post-processor for analysis of output files generated by two-dimensional **GamBet** runs. The run has two-dimensional symmetry if the command **GFile2D** was used to load a conformal triangular mesh with planar or cylindrical symmetry. In this case, **GamBet** creates an output file with a name of the form `RunName.G2D` with dose deposited in the triangular elements.

**GBView2** automatically sets appropriate units for planar or cylindrical symmetry with the following conventions:

- In a planar simulations there is no variation in the  $z$  direction. Primary particles represent a planar beam with input current in A/m and flux in photons/s/m. Dose is deposited uniformly along  $z$  (*i.e.*, the  $z$  coordinates of individual particle orbits are ignored).
- In a cylindrical simulation there is no variation in the  $\theta$  direction. Primary particles represent an annular distribution of radius  $r = \sqrt{x^2 + y^2}$  with input current in A and flux in photons/s. Dose is deposited uniformly in  $\theta$  at the  $r$  and  $z$  coordinates of particles, independent of the relative values of  $x$  and  $y$ .

### 14.1 File menu

#### LOAD DOSE FILE

Load a file with a name of the form `RunName.G2D` containing information on deposited dose in the elements of the geometric mesh.

#### LOAD TRAJECTORY FILE

This command is active if a dose file has been loaded. When a trajectory file is loaded, you can superimpose plots of primary and secondary particle orbits. Trajectory files have names of the form `RunName.GPL` and are created if the *PlotOn* command appears in the script.

#### CLOSE TRAJECTORY FILE

Close the file and remove particle orbits from plot.

#### OPEN DATA LISTING FILE

You can save the results of dose calculations performed in **GBView2** in a text file. Supply a file prefix *FPrefix* in the dialog. The output file has a name of the form `FPrefix.DAT`.

#### CLOSE DATA LISTING FILE

Terminate input to the data file. The file should be closed before opening another file or editing the present file with another Windows program.

## RUN SCRIPT

With scripts you can automatically perform complex or repetitive calculations on a set of similar solutions. This command displays a dialog with a list of script files that you have created with the suffix `SCR`. Pick a file and click OK. The script file can load data files, open and close history files, and perform any of the quantitative analysis functions. Note that the script file should be in the same directory as the data files. Section 14 describes script commands.

## CREATE SCRIPT

This command allows you to create script files using the internal editor of the post-processor. A dialog box requests a file prefix. The resulting script file will be saved as `FPREFIX.SCR`. Next, the program opens the file in the editor and writes a list of allowed commands for reference. This list follows the *EndFile* command and will be ignored by **GBView2** or **GBView3**. Enter analysis commands above the `EndFile` command. After saving the file, you can run it using the `Run script` command.

## EDIT SCRIPT

Use this command to change an existing script file. The dialog lists files in the current directory with the subscript `SCR`. Changing directories does not change the working directory of the program.

## EDIT DATA FILE

Use this command to view or to modify files with names of the form `FPREFIX.DAT`.

## EDIT FILE

Use this command to view or to modify any text file.

## 14.2 Spatial plots menu

### PLOT TYPE

The postprocessors can create a variety of colored screen and hardcopy plots.

**Mesh:** outline of the elements in the computational mesh.

**Region:** computational mesh with elements color-coded by region number.

**Contour:** lines that follow constant values of dose.

**Element:** elements of the solution space color-coded according to dose. You can activate or deactivate element boundaries with the *Toggle element outline* command.

**Surface:** a three-dimensional plot where height is proportional to dose over a region in the  $x$ - $y$  or  $z$ - $r$  plane. The spatial limits of the plot correspond to the current view window for *Mesh*, *Region*, *Contour*, or *Element* plots.

## PLOTTED QUANTITY

There are eight plot quantities: the total dose, the total energy and dose contributions from primary and secondary electrons, photons and positrons.

## PLOT LIMITS

In the default autoscale mode **GBView2** picks limits in *Contour*, *Element* and *Surface* plots that span the full range of dose. With this command you can set manual limits. In the dialog uncheck the *Autoscale* box and fill in minimum and maximum values.

## TOGGLE GRID

A set of dashed grid lines in  $x$ - $y$  or  $z$ - $r$  may be superimposed on *Mesh*, *Region*, *Contour* and *Element* plots. **GBView2** automatically chooses intervals and positions so that lines occur at convenient values of  $x$  or  $y$  (for example, 0.01 rather than 0.01153). Grids corresponding to the axes ( $x = 0.0$  or  $y = 0.0$ ) are plotted as solid lines.

## SET CONTOUR PLOT STYLE

This command is active only when the current plot type is *Contour*. There are four choices: monochrome, monochrome with labels, colored and colored with labels. In the colored mode, the lines are color-coded according to the value of the plotted quantity. A legend is included in the information window to the right of the plot. In the labeled modes, contour lines are numbered according to their values. Overlapping labels on closely-spaced lines may look better in a zoomed view.

## TOGGLE ELEMENT OUTLINE

This command determines whether the element boundaries are included in *Region* and *Element* plots. It may be necessary to deactivate outlines to view large meshes.

## TOGGLE MOUSE/KEYBOARD

By default the program uses interactive mouse entry of coordinates for commands like *Line scan* and *Zoom*. This command switches the program between mouse and keyboard input. Enter keyboard coordinates in the distance units defined by *DUnit*. In other words, if the solution program has  $DUnit = 1.0 \times 10^6$ , enter dimensions in microns.

## TOGGLE SNAP MODE

The mouse snap mode is a convenient feature of **GBView2**. When snap mode is active, the mouse returns the coordinate values closest to an integer multiple of the quantity *DSnap*. In other words, if  $DSnap = 0.5$  and the mouse position is [5.4331,-2.6253], the returned coordinates are [5.5,-2.5]. By default, snap mode is *ON*. Snap mode is automatically turned off for coordinate input to the commands **Point calculation** and **Element properties**. Otherwise, the program could pick a location closest to the snap point rather than the point at the tip of the mouse arrow, giving misleading information.

## SET DSNAP

Set the distance scale for the mouse snap mode.

The following commands change the view limits in *Mesh*, *Region*, *Contour* and *Element* plots. The current view limits of the two-dimensional plots are used when creating three-dimensional surface plots.

## ZOOM WINDOW

You can zoom in on any area of the plot by specifying the two corners of a view box with the mouse. The coordinates are displayed in the window below the plot. The returned coordinate values depend on whether snap mode is active. You can also enter view coordinates from the keyboard by issuing the `Toggle mouse/keyboard` command.

## ZOOM IN

Magnifies the view about the center of the current plot.

## EXPAND VIEW

Expands the view area about the center of the current plot.

## GLOBAL VIEW

Returns the plot boundaries to the full solution area.

## PAN

Moves the center of the plot. Enter two points to define a displacement vector.

The following commands control the appearance of surface plots. The commands are active only when a surface plot is displayed. You may notice that there is a delay if you choose the *Surface* option in the plot type command or if you change the plotted quantity when a *Surface* plot is active. To create the plot, **GBView2** must map the current quantity to a rectangular grid, performing a large number of interpolations.

## ROTATE 3D IMAGE

Rotate the *Surface* plot  $90^\circ$  in the spatial plane.

## VIEW ANGLE 3D

Change the angle of view for the *Surface* plot.

## SET GRID 3D

To create the plot, the dose is mapped to a rectangular grid with dimensions  $N_x, N_y$ . These numbers also determine the total number of grid lines in the *Surface* plot. The default values are  $N_x = N_y = 75$ . You can change values with this command. Although higher numbers give

plots with more detail, the regeneration time is longer and the screen display may be unattractive.

The next set of commands is used to send the current plot to a printer or file.

### **DEFAULT PRINTER**

Spatial and scan plots may be ported to any installed Windows printer (including network printers, postscript drivers,...). In response to the command, the current screen plot is sent to the default Windows printer. If necessary, change the default printer using the **Settings** command of Windows before running **GBView2**.

### **SAVE PLOT FILE**

Use this command to create a graphics file of the current plot in either Windows Bitmap (BMP) or Portable Network Graphics (PNG) formats. In the dialog, specify the format, the size in pixels and the file prefix. The graphics file is created in the current directory.

### **COPY TO CLIPBOARD**

Copy the current plot to the clipboard in Windows MetaFile format. The plot can then be pasted into a compatible graphics program.

## **14.3 Dose analysis menu**

The commands in the *Dose Analysis* menu generate numerical data. Most of the functions require coordinate input from the user, usually through the mouse. Therefore, the analysis menu is active only when a *Mesh*, *Region*, *Contour* or *Element* plot is displayed.

### **POINT CALCULATION**

To find the dose in an element, click on the command and then point to any position. Note that snap mode is turned OFF for coordinate input. **GBView2** displays dose values for the element that contains the point and also records information if a data file is open. To enter point coordinates by keyboard, use the **Toggle mouse/keyboard** command.

### **LINE SCAN**

The line scan is one of the most useful functions of **GBView2**. After clicking on the command, supply two points by left-clicking the mouse in a view of a *Mesh*, *Region*, *Contour* or *Element* plot to define a scan line. The snap mode is useful in this application (for example, you may want a scan to extend from 0.000 to 5.000 rather than 0.067 to 4.985.) The program determines a series of dose values in elements at equal intervals along the line. The information is recorded if a history file is open. The program also makes a screen plot of the currently-selected quantity versus distance along the scan and activates the commands in the *Scan plot* menu. **GBView2** adds fiducial lines to the plot using intelligent grid selection. This means that the plot is adjusted to fill the screen and grids are drawn at useful intervals (like 0.05 or 2.00).

## SCAN PLOT QUANTITY

There are seven plot quantities: the total dose and dose contributions from primary and secondary electrons, photons and positrons.

## SET NUMBER OF SCAN POINTS

This command sets the number of line scan points used for the screen plot and recorded in the history file. The default value is 50 and the maximum number is 500.

## ELEMENT PROPERTIES

Pick a triangular element with the mouse (or keyboard) and the post-processor writes material and dose properties of the element to the screen.

## REGION PROPERTIES

To see physical properties associated with a region of the solution space, click the mouse close to any arc or line vector of the region. Partial results are shown on the screen and a complete analysis is included in the data file.

## VOLUME INTEGRALS

No input is needed for this command. **GBView2** automatically computes integrals of quantities over the full solution volume and over individual regions. Information is recorded on the screen or in a history file.

## MATRIX FILE

This command is useful if you want to write your own analysis routines or port results to mathematical software. In response to the command, **GBView2** performs interpolations over a specified box region on a regular grid of values. It is much easier to use results in this form than to deal with the conformal triangular mesh directly. Clicking on *Matrix file* calls up a dialog. Specify the dimensions of the box along the axes (in units set by *DUnit*) and the number of calculation intervals. To illustrate, calculations are performed at positions  $X_n$  with coordinates given by

$$X_n = X_{min} + n(X_{max} - X_{min})/N_x,$$

where  $n = 0, 1, 2, \dots, N_x$ . You can also specify an output file prefix. The text file is created in the format shown below with suffix **MTX**. Units of dose depend on the mode of the **Gambet** simulation.

Matrix of values from data file Ion\_Chamber.G2D

```
ZMin:  0.000E+00   RMin:  0.000E+00
ZMax:  1.000E+01   RMax:  5.000E+00
NZ:   10    NR:   10
      Z           R       NReg   Dose (Gy)
0.000000E+00  0.000000E+00    1  4.409377E+06
1.000000E+00  0.000000E+00    1  4.143115E+06
2.000000E+00  0.000000E+00    1  3.733472E+06
3.000000E+00  0.000000E+00    1  1.859380E+06
4.000000E+00  0.000000E+00    1  1.215571E+06
5.000000E+00  0.000000E+00    1  6.421903E+05
...
```

## SMOOTH DOSE

Statistical noise is invariably associated with Monte Carlo calculations. The interactive plotting and calculation functions of **GBView2** make it easy to visualize and to quantify statistical variations of dose. These variations may be significant in solutions where the mesh has a large number of small elements. With this command you can apply averaging to reduce variations between elements at the expense of some loss of spatial resolution. The following procedure is used. The program cycles through *NSmoothParam* cycles. In each cycle, the element dose is set equal to 0.75 times the previous dose plus 0.25 times a volume-weighted average of the doses in the three adjacent elements. You can apply this command several times to achieve the desired level of smoothing. If smoothing becomes excessive, reload the original file.

## SMOOTH PARAMETER

Set the value of *NSmoothParam*. Larger values give stronger smoothing. The default value is 10.

## SAVE DOSE FILE

You can save a smoothed distribution in a standard two dimensional dose file. Supply a file prefix *FPrefix*. Be sure to use a different prefix if you want to preserve the original data. The file is saved in the current directory under the name *FPrefix.G2D*.

## 14.4 Scan plot menu

The commands of the `Scan plot` menu become active when a plot is created following the line scan command.

## OSCILLOSCOPE MODE

In the oscilloscope mode, a scan plot assumes characteristics of a digital oscilloscope. The program superimposes a cross-hair pattern on the graph. Plot values at the intersection are displayed in the information window. Move the marker along the plot by moving the mouse. If you click the left mouse button at a point, the program displays the plot values along with the

numerical derivative and integral of the curve. The definite integral is taken from the left-hand side of the plot to the current point. Values are displayed on the screen and written to the data file if open. Press the right mouse button to exit the oscilloscope mode.

### **TOGGLE SCAN SYMBOLS**

The setting determines whether symbols are added to the scan plot to show calculated points.

### **TOGGLE GRID**

The setting determines whether grid lines are added to the screen and hardcopy scan plots.

### **CLOSE SCAN PLOT**

The scan plot must be closed before you can continue to use the *File* and *Analysis* functions of **GBView2**. This command closes the scan plot and returns the program to the previous spatial plot.

---

## 15 GBView3

**GBView3** is an interactive graphical post-processor for analysis of output files generated by three-dimensional **GamBet** runs. The run has three-dimensional symmetry if the command **GFile3D** was used to load a conformal hexahedron mesh. In this case, **GamBet** creates an output file with a name of the form `RunName.G3D` with dose deposited in the hexahedron elements.

### 15.1 File menu

**GBView3** has the following popup menus: **File operations**, **Analysis**, **Plane plots**, **Slice plots**, **Surface plots** and **Help**. When the program starts, only the **File** and **Help** menus are active. You must load a data file in order to create plots or to perform analyses. We shall first review options in the **File** menu.

#### LOAD DOSE FILE

In response to the command, the program displays a dialog with a list of files with names of the form `FPREFIX.G3D`. Changing the directory in the dialog changes the program working directory. Pick an available file and click *OK*. The program loads the solution and updates the status bar. If data retrieval is successful, the analysis and plot menus become active.

#### DOSE FILE INFORMATION

Display a dialog with information on the currently-loaded dose file.

#### LOAD TRAJECTORY FILE

This command is active when a dose file is active. When a trajectory file is loaded, you can superimpose plots of primary and secondary particle orbits. Trajectory files have names of the form `RunName.GPL` and are created if the **PLOTON** command appears in the script.

#### CLOSE TRAJECTORY FILE

Close the file and remove particle orbits from plot.

#### OPEN DATA LISTING FILE

You can save the results of dose calculations performed in **GBView3** in a text file. Supply a file prefix `FPrefix` in the dialog. The output file has a name of the form `FPrefix.DAT`.

#### CLOSE DATA LISTING FILE

Terminate input to the data file. The file should be closed before opening another file or editing the present file with another Windows program.

## **RUN SCRIPT**

With scripts you can automatically perform complex or repetitive calculations on a set of similar solutions. This command displays a dialog with a list of script files that you have created with the suffix **SCR**. Pick a file and click *OK*. The script file can load data files, open and close history files, and perform any of the quantitative analysis functions described in this manual. Note that the script file should be in the same directory as the data files. Section 14 describes script commands.

## **CREATE SCRIPT**

This command allows you to create scripts using the internal editor of the post-processor. A dialog box requests a file prefix. The resulting script will be saved as **FPREFIX.SCR**. Next, the program opens the file in the editor and writes a list of allowed commands for reference. This list follows the **EndFile** command and will be ignored by **GBView2** or **GBView3**. Enter analysis commands above the **EndFile** command. After saving the file, you can run it using the **Run script** command.

## **EDIT SCRIPT**

Use this command to change an existing script. The dialog lists files in the current directory with the subscript **SCR**. Changing directories does not change the working directory of the program.

## **EDIT DATA FILE**

Use this command to view or to modify files with names of the form **FPREFIX.DAT**.

## **EDIT FILE**

Use this command to view or to modify any text file.

## **15.2 Dose analysis menu**

The commands in the **Dose analysis** menu generate numerical data. Some functions operate only if a data listing file has been opened.

### **POINT CALCULATION**

To find the dose at a position, supply  $(x, y, z)$  coordinates in the dialog in units set by **DUnit** in the **GamBet** run. **GBView3** displays writes dose values for the element that contains the point and also records information if a data file is open.

### **LINE SCAN**

In response to this command, **GBView3** calculates a series of dose values along a specified line. Supply  $(x, y, z)$  coordinates of the start and end points. The information is recorded if a history file is open. The program also makes a screen plot of the currently-selected quantity versus distance along the scan and activates the commands in the **Scan plot** menu. **GBView3** adds

fiducial lines to the plot using intelligent grid selection. This means that the plot is adjusted to fill the screen and grids are drawn at useful intervals (like 0.05 or 2.00).

### **SCAN PLOT QUANTITY**

There are seven plot quantities: the total dose, the total energy and dose contributions from primary and secondary electrons, photons and positrons.

### **SET NUMBER OF SCAN POINTS**

This command sets the number of line scan points used for the screen plot and recorded in the history file. The default value is 50 and the maximum number is 500.

### **REGION PROPERTIES**

To see physical properties associated with a region of the solution space, supply the region number in the dialog. Partial results are shown on the screen and a complete analysis is included in the data file.

### **VOLUME INTEGRALS**

No input is needed for this command. **GBView3** automatically computes integrals of dose over the full solution volume and over individual regions. Information is recorded on the screen or in a history file.

### **CREATE MATRIX FILE**

This command is useful if you want to write your own analysis routines or port results to mathematical software. In response to the command, **GBView3** performs interpolations over a specified box region on a regular grid of values. It is much easier to use results in this form than to deal with the conformal hexahedron mesh directly. Clicking on **Create matrix file** calls up a dialog. Specify the dimensions of the box along each axis (in units set by **DUnit**) and the number of calculation intervals. To illustrate, calculations are performed at positions  $X_n$  with coordinates given by

$$X_n = X_{min} + n(X_{max} - X_{min})/N_x,$$

where  $n = 0, 1, 2, \dots, N_x$ . You can also specify an output file prefix. The text file is created in the format shown below. Units of dose depend on the mode of the **Gambet** simulation.

GBView3 matrix file: OutMatrix.MTX

```
XMin: -7.50000E+00  XMax:  7.50000E+00  NX:    10
YMin: -7.50000E+00  YMax:  7.50000E+00  NY:    10
ZMin:  0.00000E+00  ZMax:  1.50000E+01  NZ:    10
```

X	Y	Z	Energy	Density	NReg
-7.50000E+00	-7.50000E+00	0.00000E+00	5.93616E+04	1.00000E+00	1
-6.00000E+00	-7.50000E+00	0.00000E+00	5.93616E+04	1.00000E+00	1
-4.50000E+00	-7.50000E+00	0.00000E+00	5.93616E+04	1.00000E+00	1
-3.00000E+00	-7.50000E+00	0.00000E+00	5.93616E+04	1.00000E+00	1
-1.50000E+00	-7.50000E+00	0.00000E+00	5.93616E+04	1.00000E+00	1
...					

## SMOOTH DOSE

This command has the same function of the `Smooth dose` command in **GBView2**. You can apply the command several times to achieve the desired level of smoothing. If smoothing becomes excessive, reload the original file.

## SMOOTH PARAMETER

Set the value of `NSmoothParam`. Larger values give stronger smoothing. The default value is 5.

## SAVE DOSE FILE

You can save a smoothed distribution in a standard three-dimensional dose file. Supply a file prefix `FPrefix`. Be sure to use a different prefix if you want to preserve the original data. The file is saved in the current directory under the name `FPrefix.G3D`.

## 15.3 Plane plots

Plane plots are two-dimensional plots that show the variation of dose over a plane normal to one of the Cartesian axes. Plane plots provide simple and quick views of the solution space. The technique is to generate a set of values on a rectangular mesh over a specified planar region and then to create plots in a variety of styles. No attempt is made to connect the plot mesh with the conformal mesh of the simulation. Slice plots are also two-dimensional in a plane normal to a Cartesian axis. The difference is that slice plots are built using the computational mesh. They show precise region boundaries and contour lines in true scale. Plot options are more limited because slice plots are more difficult to construct than plane plots.

## SET PLANE

This command brings up a dialog to set the plane for the plot. Set the normal axis with the radio buttons at the top. For example, for a choice of  $z$  the plot will be created in the  $x$ - $y$  plane. You can use the slider bar to set the position along the normal axis or type a value in the box. The range of the slider bar is automatically set to the limits of the solution volume along

the normal axis. The boxes at the bottom determine the plot range in the normal plane. The default settings are the limits of the solution volume. Note that plane plots are constructed to fill the maximum screen area. They do not preserve scaling in the normal plane.

The following commands are in the PLOT CONTROL popup menu:

### **PLOT STYLE**

The command brings up a dialog to set the plot style. The numbers at the bottom give the resolution of the mesh used to create the plot. Higher values give more detail but require longer regenerate times. The default is a  $75 \times 75$  mesh.

### **PLOT QUANTITY**

The single allowed quantity is the total dose.

### **PLOT LIMITS**

Set limits for the plotted quantity. When Autoscale is active, **GBView3** automatically sets limits based on the range of values in the solution file.

### **ROTATE PLOT**

This command is active only for the **Filled contours 3D** and **Gradient plot 3D** plot styles. You can rotate the plot in 90 degree increments for the best view.

The commands of the **EXPORT PLOT** menu are used to generate hardcopy or to create plot files.

### **DEFAULT PRINTER**

Use this command to send the current plot to any installed Windows printer (including network printers, postscript drivers, PDF drivers...). Note that the destination is the default Windows printer. If necessary, change the default using the **Settings** command of Windows before issuing the command.

### **SAVE PLOT FILE**

Use this command to create a graphics file of the current plot in either Windows Bitmap (BMP) or Portable Network Graphics (PNG) formats. In the dialog, specify the format, the size in pixels and the file prefix. The graphics file is created in the current directory.

### **COPY TO CLIPBOARD**

The current plot is copied to the clipboard in Windows Metafile format. You can then paste the image into graphics software.

## 15.4 Slice plots

Slice plots are two-dimensional plots that show the variation of quantities over a plane normal to one of the Cartesian axes. In contrast to plane plots, slice plots are based on the structure of the mesh projected to a slice plane. This structure may be quite complex for a conformal mesh; therefore, slice plots require more computational effort. To facilitate the process, slices are at discrete locations along the normal axis corresponding roughly to the planes of the **MetaMesh** foundation mesh. The precise rendering of spatial information allow point-and-click analysis operations (point calculation, line scan, ...) in the slice.

The **Change view** popup menu contains commands to set the slice plane and to adjust the dimensions of the plot.

### **SET SLICE PLANE PROPERTIES**

This command calls up the same dialog as the **Set plane** command in the **Plane plot** menu. You can change the normal axis, change the position along the normal axis, and set plot limits in the normal plane.

### **SLICE NORMAL TO X**

### **SLICE NORMAL TO Y**

### **SLICE NORMAL TO Z**

Quick commands to change the normal axis.

### **JUMP FORWARD**

### **STEP FORWARD**

### **STEP BACKWARD**

### **JUMP BACKWARD**

Move along the slice axis by small or large steps. The small step is approximately one layer of the foundation mesh and the large step is 5 layers. The term forward implies motion toward higher indices of the normal axis. The slider bar in the orientation area to the right of the plot shows the current location.

### **ZOOM WINDOW**

As an alternative to the entries in the **Set slice plane** dialog, you can interactively change plot limits in the normal plane using the mouse. Choose the command and move the mouse pointer into the plot area. The status bar enters coordinate mode. It shows the current mouse position in the plot and the snap mode. Use the left button to pick one corner and then move the mouse to create a view box. Click the left button again, and the plot regenerates. On any coordinate operation, press the **F1** key if want to enter values from the keyboard. Note that the normal plane box in the orientation area to the right of the plot shows the dimensions of the slice plane and the outline of the current zoomed view.

### **ZOOM IN**

Enlarge the plot about the current view center.

## EXPAND VIEW

Expand the plot about the current view center.

## GLOBAL VIEW

Enlarge the plot boundaries to show the entire normal plane.

## PAN

When the plot is zoomed, you can use this operation to shift the current view center. Use the mouse to define a relative starting and ending point for the shift.

The commands in the `Plot control` popup menu are used to set the plot style and mouse options.

## SET SLICE PLOT STYLE

This command brings up the dialog to choose the plot style. The *Region plot* style is a cross-section view of the mesh element divisions color-coded by region. In contrast to the logical plane plot of **MetaMesh**, **GBView3** attempts to resolve the exact mesh structure in the plane. The *Contour plot* style shows lines of constant dose, and *Element plot* shows element cross-sections color-coded by dose. The plot option *Region shading* applies to contour plots. When active, **GBView3** adds light background colors to show the boundaries between different material regions. The *Element outline* option applies to element plots. When active, **GBView3** adds the boundaries of elements. The check boxes on the right-hand side control the data included in plots. The *Electron*, *Photon* and *Positron* options are active only if a trajectory plot has been loaded.

## PLOT LIMITS

In the autoscale mode, **GBView3** chooses defaults for the minimum and maximum potential values of dose. Deactivate autoscale to set the values manually.

## NUMBER OF CONTOUR LINES

Change the number of lines for contour plots.

## TOGGLE SNAP MODE

Mouse coordinates for commands such as `Zoom window`, `Pan`, and `Scan in slice` can be entered in two modes. In the normal mode, the returned position corresponds to the mouse position on the screen. In the snap mode, the program picks a point at an even interval close to the mouse position. The returned point depends on the value of the parameter *DSnap*. For example if  $DSnap = 0.1$  and the mouse is at position (6.2345,-5.6113), the returned position is (6.2000,-5.6000). The status bar displays the actual or snapped position of the mouse.

## SET SNAP DISTANCE

Change the value of *DSnap* from the default value determined by the program.

## TOGGLE GRID

A set of dashed grid lines can be superimposed on slice plots. **GBView3** automatically chooses intervals and positions so that the lines occur at convenient values along the horizontal and vertical directions (for example, 0.01 rather than 0.01153). Listings of the grid intervals are included in the axis labels. Grids corresponding to the normal plane axes are plotted as solid lines.

## 15.5 Analysis in a slice

You can determine dose values at points and along scan lines within the slice using the commands of the **Analysis** popup menu. Results are recorded if a data file is open.

### POINT CALCULATION

This command is useful to make quick checks of dose values in the solution volume. After you click the **Point calculation** command, move the mouse into the plot area. The mouse pointer changes to a cross-hair pattern and the status bar enters coordinate mode. Click the left button to specify a point or press the **F1** key to enter the coordinates from the keyboard. Note that mouse coordinates will shift between discrete values if snap mode is active. **GBView3** calculates dose values at the point in the normal plane given by the coordinates.

### LINE SCAN

The line scan is one of the most useful **GBView3** capabilities. After clicking on the command, supply two points with the mouse to define a scan line (or press the **F1** key to enter coordinates manually). The snap mode is useful in this application (for example, you may want the scan to extend from 0.000 to 5.000 rather than 0.067 to 4.985.) The program computes a series of values of field quantities in the normal plane at equal intervals along the line. Complete information is recorded if a data file is open. The program also makes a screen plot of the currently-selected quantity versus distance along the scan and activates the **Scan plot** menu. **GBView3** adds fiducial lines to the plot using intelligent grid selection. This means that the plot is adjusted to fill the screen and grids are drawn at useful intervals (like 0.05 or 2.00).

### SET SCAN QUANTITY

There are seven plot quantities: the total dose and dose contributions from primary and secondary electrons, photons and positrons.

### SET NUMBER OF SCAN POINTS

This command sets the number of line scan points in the screen plot and data file listing. The default value is 50 and the maximum number is 250.

In addition to the standard **Export plot** options, the **Scan plot** menu contains the following command:

### OSCILLOSCOPE MODE

In oscilloscope mode, a scan plot assumes characteristics of a digital oscilloscope. **GBView3**

superimposes a cross-hair pattern on the graph. Plot values at the intersection are displayed in the information window. Move the marker along the plot by moving the mouse. If you click the left mouse button at a point, the program records information when a data file is open. Press the right mouse button to exit the oscilloscope mode.

Click *Return* to exit the scan plot and to return to the slice plot.

## 15.6 Surface plots

Surface plots are three-dimensional views of the solution space. Three types of information may be superimposed: 1) boundaries of regions with foreground lighting, 2) dose values in a slice plane normal to one of the Cartesian axes and 3) dose values of region surface facets. Surface plots are created from the conformal mesh and preserve true spatial scaling.

View controls are the same as those in **Geometer** and **Metamesh**. With the mouse cursor in the center, the left button moves the viewpoint inward (zoom) and the right outward (expand view). Changes are reflected in the orientation box in the upper-right portion of the screen. The main plot is updated when you release the mouse button. With the cursor on the right-hand side of the plot, press the left button to walk around the object and the right button to move in the direction of the cursor arrow.

The commands of the *Plot control* popup menu control the appearance of the plot.

### **SURFACE PLOT STYLE**

This command brings up a dialog with the following options: *Hidden surface/Wireframe* (plot style for region boundary facets), *Include facets* (include or omit the borders of region boundary facets), *Include reference grid* (plot lines with tick marks along the edges of the solution volume) and *Include surface plot* (include a plot of dose in a plane normal to one of the Cartesian axes). If a normal plane dose plot is active, you can use the controls on the right-hand side to set the normal direction of the plane and its position along the normal axis.

### **DISPLAYED REGIONS**

The command brings up a dialog where you can pick region boundaries to superimpose on a plot. The default is that no regions are displayed. You can also determine whether the region boundary is plotted as a perspective view with foreground lighting or with color-coding according to the average local dose. Note that the program issues an error message in the normal plane plot is deactivated and no regions are chosen.

### **SET CUT PLANES**

In a hidden surface plot, internal details may be obscured by surrounding parts or regions. This command brings up a dialog that allows you to adjust the display areas along the  $x$ ,  $y$  and  $z$  axes. **GBView3** does not display facets that lie outside the limits. With this feature you can create cutaway views. The default is that cut limits are set equal to the dimensions of the solution volume so that all facets are included.

## SET SURFACE PLOT QUANTITY

There are seven available surface quantities: the total dose and dose contributions from primary and secondary electrons, photons and positrons.

## 15.7 Saving and loading views

The creation of plots for presentations may involve some effort. With the following two commands, you can save all the current view parameters in **GBView3** and immediately restore the plot.

### SAVE NAMED VIEW

Save the view parameters for the current plot. Quantities such as the slice axis, slice position and zoom limits are saved for two dimensional plots. Parameters for three-dimensional plots include the viewpoint position, displayed regions and cut planes. The information is stored in a text file in the current directory with a name of the form **FPREFIX.FPV**.

### LOAD NAMED VIEW

Load a view file and refresh the plot

The file contains the complete set of plot parameters. This excerpt illustrates the format:

```
Program: GBView3
2D/3D: 2D
DisplayBy: Regions
Outline: On
NSlice:    40
SliceAxis: YAxis
PlotType: LogElemUp
XPMin: -1.500000E+00
XPMax:  4.250000E+00
...
```

If a specific solution file is loaded, the plot will be restored exactly. The saved view feature in **GBView3** has two useful features if a different meshes are loaded:

- Dynamic adaptation to different solutions.
- Option for user control of the view parameter set.

Regarding the first feature, there are situations where you want to create consistent views of a set of solutions with different geometries, maintaining a similar appearance. Some plot properties (like the viewpoint rotation matrix) are applicable to any solution, but others (like region cut planes or slice plot limits) depend on the geometry. **GBView3** checks each plot parameter for validity. If a parameter is outside the allowed range for the currently-loaded solution, the program computes an alternative. The goal is to preserve as many features of the view as possible.

You can modify view files with an editor. The order of entries is not rigid. On input, **GBView3** uses a free-form parser. If a parameter is missing, the program simply makes no change from the value current in the program. The implication is that you can modify a saved view to include only elements essential to your application. For example, you could compare a series of assemblies with different sizes, maintaining an orthographic 3D view from the same point in Cartesian space. In this case, the view file would contain only the entries:

```
DView: 1.000000E+37
R11: 8.660253E-01
R12: -5.000002E-01
R13: 0.000000E+00
R21: 1.669031E-01
R22: 2.890846E-01
R23: 9.426408E-01
R31: -4.713208E-01
R32: -8.163510E-01
R33: 3.338061E-01
```

---

## 16 GBView2 and GBView3 script operation

Scripts to control **GBView2** or **GBView3** have names of the form `FPREFIX.SCR`. They should be in the same directory as the data files. Scripts are text files that follow the **Gambet** syntax conventions. The program ignores blank lines and indentations. Data lines use the standard delimiters and comment lines begin with `'*` (asterisk). Processing stops when the `ENDFILE` command is encountered.

To execute a script, click on the *Run script* command in the *File* menu. The program shows a list of available script files. Pick a file and click OK. The script operates on the presently-loaded data file or you can load other files from within the script. You can also sequentially open one or more data files.

**GBView2** and **GBView3** can run autonomously under script file control from the Windows command prompt. Suppose you have a file `GTest.SCR` in the directory `\GAMBET\BUFFER` and that `GBVIEW2.EXE` is in the directory `\GAMBET`. From `\GAMBET\BUFFER`, type

```
../GBVIEW2 GTEST <Enter>
```

The following commands may appear in a script:

### INPUT BACKSCAT.G2D

Close the current data file and load a file for analysis. The parameter is the full name of the data file. For the command illustrated, the post-processor would load the file `BackScat.G2D`. You can load several files for sequential analysis.

### OUTPUT BACKSCAT

Close the current history file and open an output file `BACKSCAT.DAT`.

### POINT 5.65 10.68

### POINT 0.00 0.00 10.00

Calculate the dose at the specified point and write the results to the data file. In **GBView2**, the two real number parameters are the coordinates of the point ( $x$ - $y$  or  $z$ - $r$ ) in units set by *DUnit*. **GBView3** requires three real numbers for the  $x$ ,  $y$ ,  $z$  coordinates.

### SCAN 0.00 0.00 10.00 0.00

### SCAN (0.00, 0.00, 3.00) (0.00, 0.00, 10.00)

Write calculated dose values on a line between the specified points to the data file. In **GBView2**, the four real-number parameters are the starting and end coordinates in units set by *DUnit*: [ $xstart$   $ystart$   $xend$   $yend$ ] or [ $zstart$   $rstart$   $zend$   $rend$ ]. In **GBView3**, supply six real-number parameters, [ $xstart$   $ystart$   $zstart$   $xend$   $yend$   $zend$ ].

### PATH PathName

### PATH LargeArc.DAT

Calculate the dose at the coordinates listed in the file *PathName* and write the results to the data file. The path file must be available in the current directory. For **GBView2** each data line contains

two real numbers giving a position  $(x, y)$  or  $(z, r)$  (depending on the symmetry of the geometry file). For **GBView3** the data lines should contain three real numbers giving  $(x, y, z)$  values. Enter the coordinates in units set by *DUnit*. The path file may contain any number of data lines, blank lines and comments (starting with an asterisk). End the list with the *EndFile* command.

#### **ELEMENT 5.65 10.68**

#### **ELEMENT 0.00, 1.00, 5.67**

Write the properties of the element at the specified point to the data file. In **GBView2**, the two real number parameters are the coordinates of the point  $(x-y$  or  $z-r)$  in units set by *DUnit*. Enter three real numbers  $(x, y, z)$  in **GBView3**.

#### **NSCAN 150**

Set the number of points in a line scan. The default is 50 and the maximum number is 500.

#### **REGION 5**

Write volume integrals for a region to the data file. The integer parameter is the region number.

#### **VOLUMEINT**

Write volume integrals for the full solution and regions to the data file.

#### **MATRIX GOLDFOIL 10 20 0.00 0.00 5.00 10.00**

#### **MATRIX GOLDFOIL 0.00 1.00 10 0.00 1.00 10 0.00 5.00 25**

Create a file and record dose values on a regular rectangular matrix. In **GBView2**, the command must contain the following seven parameters.

Param 1: The prefix of the matrix file *FPREFIX.MTX* (string).

Param 2: Number of intervals along the  $x$  (or  $z$ ) direction (integer).

Param 3: Number of intervals along the  $y$  (or  $r$ ) direction (integer).

Param 4-7: The coordinates of the corners of a box in the solution volume,  $[x_1, y_1, x_2, y_2]$  or  $[z_1, r_1, z_2, r_2]$  (real).

**GBView3** requires the following 10 parameters:

Param 1: The prefix of the matrix file *FPREFIX.MTX* (string).

Param 2-4: The box dimensions  $x_{min}$  and  $x_{max}$  (real) and the number of intervals  $N_x$  along  $x$  (integer).

Param 5-7: The box dimensions  $y_{min}$  and  $y_{max}$  (real) and the number of intervals  $N_y$  along  $y$  (integer).

Param 8-10: The box dimensions  $z_{min}$  and  $z_{max}$  (real) and the number of intervals  $N_z$  along  $z$  (integer)

#### **ENDFILE**

Terminate execution of the script file. You can add descriptive text in any format after this command.

---

## 17 Penelope predefined materials

This section lists predefined materials included in the Penelope file PDCOMPOS.P06 file along with their identifying numbers. These material numbers may be used in the

MATERIAL MatNo [CONDUCTOR, INSULATOR]

command of the *COMPOSITION* section.

### 17.1 Elements

For elements, the material number equals the atomic number

1	Hydrogen	50	Tin
2	Helium	51	Antimony
3	Lithium	52	Tellurium
4	Beryllium	53	Iodine
5	Boron	54	Xenon
6	Amorphous carbon	55	Cesium
7	Nitrogen	56	Barium
8	Oxygen	57	Lanthanum
9	Fluorine	58	Cerium
10	Neon	59	Praseodymium
11	Sodium	60	Neodymium
12	Magnesium	61	Promethium
13	Aluminum	62	Samarium
14	Silicon	63	Europium
15	Phosphorus	64	Gadolinium
16	Sulfur	65	Terbium
17	Chlorine	66	Dysprosium
18	Argon	67	Holmium
19	Potassium	68	Erbium
20	Calcium	69	Thulium
21	Scandium	70	Ytterbium
22	Titanium	71	Lutetium
23	Vanadium	72	Hafnium
24	Chromium	73	Tantalum
25	Manganese	74	Tungsten
26	Iron	75	Rhenium
27	Cobalt	76	Osmium
28	Nickel	77	Iridium
29	Copper	78	Platinum
30	Zinc	79	Gold
31	Gallium	80	Mercury
32	Germanium	81	Thallium
33	Arsenic	82	Lead
34	Selenium	83	Bismuth

35	Bromine	84	Polonium
36	Krypton	85	Astatine
37	Rubidium	86	Radon
38	Strontium	87	Francium
39	Yttrium	88	Radium
40	Zirconium	89	Actinium
41	Niobium	90	Thorium
42	Molybdenum	91	Protactinium
43	Technetium	92	Uranium
44	Ruthenium	93	Neptunium
45	Rhodium	94	Plutonium
46	Palladium	95	Americium
47	Silver	96	Curium
48	Cadmium	97	Berkelium
49	Indium	98	Californium
		99	Einsteinium

## 17.2 Compounds and mixtures

Compounds and mixtures appear in alphabetical order.

100	Acetone
101	Acetylene
102	Adenine
103	Adipose tissue (ICRP)
104	Air, dry (near sea level)
105	Alanine
106	Aluminum oxide
107	Amber
108	Ammonia
109	Aniline
110	Anthracene
111	B-100 bone-equivalent plastic
112	Bakelite
113	Barium fluoride
114	Barium sulfate
115	Benzene
116	Beryllium oxide
117	Bismuth germanium oxide
118	Blood (ICRP)
119	Bone, compact (ICRU)
120	Bone, cortical (ICRP)
121	Boron carbide
122	Boron oxide
123	Brain (ICRP)
124	Butane
125	N-butyl alcohol
126	C-552 air-equivalent plastic
127	Cadmium telluride

128 Cadmium tungstate  
129 Calcium carbonate  
130 Calcium fluoride  
131 Calcium oxide  
132 Calcium sulfate  
133 Calcium tungstate  
134 Carbon dioxide  
135 Carbon tetrachloride  
136 Cellulose acetate, cellophane  
137 Cellulose acetate butyrate  
138 Cellulose nitrate  
139 Ceric sulfate dosimeter solution  
140 Cesium fluoride  
141 Cesium iodide  
142 Chlorobenzene  
143 Chloroform  
144 Concrete, portland  
145 Cyclohexane  
146 1,2-dichlorobenzene  
147 Dichlorodiethyl ether  
148 1,2-dichloroethane  
149 Diethyl ether  
150 N,n-dimethyl formamide  
151 Dimethyl sulfoxide  
152 Ethane  
153 Ethyl alcohol  
154 Ethyl cellulose  
155 Ethylene  
156 Eye lens (ICRP)  
157 Ferric oxide  
158 Ferrobtoride  
159 Ferrous oxide  
160 Ferrous sulfate dosimeter solution  
161 Freon-12  
162 Freon-12b2  
163 Freon-13  
164 Freon-13b1  
165 Freon-13i1  
166 Gadolinium oxysulfide  
167 Gallium arsenide  
168 Gel in photographic emulsion  
169 Pyrex glass  
170 Glass, lead  
171 Glass, plate  
172 Glucose  
173 Glutamine  
174 Glycerol  
175 Graphite  
176 Guanine

177 Gypsum, plaster of Paris  
178 N-heptane  
179 N-hexane  
180 Kapton polyimide film  
181 Lanthanum oxybromide  
182 Lanthanum oxysulfide  
183 Lead oxide  
184 Lithium amide  
185 Lithium carbonate  
186 Lithium fluoride  
187 Lithium hydride  
188 Lithium iodide  
189 Lithium oxide  
190 Lithium tetraborate  
191 Lung (ICRP)  
192 M3 wax  
193 Magnesium carbonate  
194 Magnesium fluoride  
195 Magnesium oxide  
196 Magnesium tetraborate  
197 Mercuric iodide  
198 Methane  
199 Methanol  
200 Mix d wax  
201 Ms20 tissue substitute  
202 Muscle, skeletal (ICRP)  
203 Muscle, striated (ICRU)  
204 Muscle-equivalent liquid, with sucrose  
205 Muscle-equivalent liquid, without sucrose  
206 Naphthalene  
207 Nitrobenzene  
208 Nitrous oxide  
209 Nylon, du Pont elvamide 8062  
210 Nylon, type 6 and type 6/6  
211 Nylon, type 6/10  
212 Nylon, type 11 (rilsan)  
213 Octane, liquid  
214 Paraffin wax  
215 N-pentane  
216 Photographic emulsion  
217 Plastic scintillator (vinyltoluene based)  
218 Plutonium dioxide  
219 Polyacrylonitrile  
220 Polycarbonate (makrolon, lexan)  
221 Polychlorostyrene  
222 Polyethylene  
223 Polyethylene terephthalate (mylar)  
224 Polymethyl methacrylate (lucite, perspex, plexiglass)  
225 Polyoxymethylene

226 Polypropylene  
227 Polystyrene  
228 Polytetrafluoroethylene (teflon)  
229 Polytrifluorochloroethylene  
230 Polyvinyl acetate  
231 Polyvinyl alcohol  
232 Polyvinyl butyral  
233 Polyvinyl chloride  
234 Polyvinylidene chloride (saran)  
235 Polyvinylidene fluoride  
236 Polyvinyl pyrrolidone  
237 Potassium iodide  
238 Potassium oxide  
239 Propane  
240 Propane, liquid  
241 N-propyl alcohol  
242 Pyridine  
243 Rubber, butyl  
244 Rubber, natural  
245 Rubber, neoprene  
246 Silicon dioxide  
247 Silver bromide  
248 Silver chloride  
249 Silver halides in photographic emulsion  
250 Silver iodide  
251 Skin (ICRP)  
252 Sodium carbonate  
253 Sodium iodide  
254 Sodium monoxide  
255 Sodium nitrate  
256 Stilbene  
257 Sucrose  
258 Terphenyl  
259 Testes (ICRP)  
260 Tetrachloroethylene  
261 Thallium chloride  
262 Tissue, soft (ICRP)  
263 Tissue, soft (ICRU four-component)  
264 Tissue-equivalent gas (methane based)  
265 Tissue-equivalent gas (propane based)  
266 A-150 tissue-equivalent plastic  
267 Titanium dioxide  
268 Toluene  
269 Trichloroethylene  
270 Triethyl phosphate  
271 Tungsten hexafluoride  
272 Uranium dicarbide  
273 Uranium monocarbide  
274 Uranium oxide

275 Urea  
276 Valine  
277 Viton fluoroelastomer  
278 Water, liquid  
279 Water vapor  
280 Xylene

---

## 18 Penelope data files

The following files must be located in the directory /PATH/GAMBET/REFERENCE.

eeldx001.p06	eeldx046.p06	eeldx091.p06	pdebr34.p06	pdebr79.p06
eeldx002.p06	eeldx047.p06	eeldx092.p06	pdebr35.p06	pdebr80.p06
eeldx003.p06	eeldx048.p06	eeldx093.p06	pdebr36.p06	pdebr81.p06
eeldx004.p06	eeldx049.p06	eeldx094.p06	pdebr37.p06	pdebr82.p06
eeldx005.p06	eeldx050.p06	eeldx095.p06	pdebr38.p06	pdebr83.p06
eeldx006.p06	eeldx051.p06	eeldx096.p06	pdebr39.p06	pdebr84.p06
eeldx007.p06	eeldx052.p06	eeldx097.p06	pdebr40.p06	pdebr85.p06
eeldx008.p06	eeldx053.p06	eeldx098.p06	pdebr41.p06	pdebr86.p06
eeldx009.p06	eeldx054.p06	eeldx099.p06	pdebr42.p06	pdebr87.p06
eeldx010.p06	eeldx055.p06	pdatconf.p06	pdebr43.p06	pdebr88.p06
eeldx011.p06	eeldx056.p06	pdrang.p06	pdebr44.p06	pdebr89.p06
eeldx012.p06	eeldx057.p06	pdcompos.p06	pdebr45.p06	pdebr90.p06
eeldx013.p06	eeldx058.p06	pdebr01.p06	pdebr46.p06	pdebr91.p06
eeldx014.p06	eeldx059.p06	pdebr02.p06	pdebr47.p06	pdebr92.p06
eeldx015.p06	eeldx060.p06	pdebr03.p06	pdebr48.p06	pdebr93.p06
eeldx016.p06	eeldx061.p06	pdebr04.p06	pdebr49.p06	pdebr94.p06
eeldx017.p06	eeldx062.p06	pdebr05.p06	pdebr50.p06	pdebr95.p06
eeldx018.p06	eeldx063.p06	pdebr06.p06	pdebr51.p06	pdebr96.p06
eeldx019.p06	eeldx064.p06	pdebr07.p06	pdebr52.p06	pdebr97.p06
eeldx020.p06	eeldx065.p06	pdebr08.p06	pdebr53.p06	pdebr98.p06
eeldx021.p06	eeldx066.p06	pdebr09.p06	pdebr54.p06	pdebr99.p06
eeldx022.p06	eeldx067.p06	pdebr10.p06	pdebr55.p06	pdee101.p06
eeldx023.p06	eeldx068.p06	pdebr11.p06	pdebr56.p06	pdee102.p06
eeldx024.p06	eeldx069.p06	pdebr12.p06	pdebr57.p06	pdee103.p06
eeldx025.p06	eeldx070.p06	pdebr13.p06	pdebr58.p06	pdee104.p06
eeldx026.p06	eeldx071.p06	pdebr14.p06	pdebr59.p06	pdee105.p06
eeldx027.p06	eeldx072.p06	pdebr15.p06	pdebr60.p06	pdee106.p06
eeldx028.p06	eeldx073.p06	pdebr16.p06	pdebr61.p06	pdee107.p06
eeldx029.p06	eeldx074.p06	pdebr17.p06	pdebr62.p06	pdee108.p06
eeldx030.p06	eeldx075.p06	pdebr18.p06	pdebr63.p06	pdee109.p06
eeldx031.p06	eeldx076.p06	pdebr19.p06	pdebr64.p06	pdee110.p06
eeldx032.p06	eeldx077.p06	pdebr20.p06	pdebr65.p06	pdee111.p06
eeldx033.p06	eeldx078.p06	pdebr21.p06	pdebr66.p06	pdee112.p06
eeldx034.p06	eeldx079.p06	pdebr22.p06	pdebr67.p06	pdee113.p06
eeldx035.p06	eeldx080.p06	pdebr23.p06	pdebr68.p06	pdee114.p06
eeldx036.p06	eeldx081.p06	pdebr24.p06	pdebr69.p06	pdee115.p06
eeldx037.p06	eeldx082.p06	pdebr25.p06	pdebr70.p06	pdee116.p06
eeldx038.p06	eeldx083.p06	pdebr26.p06	pdebr71.p06	pdee117.p06
eeldx039.p06	eeldx084.p06	pdebr27.p06	pdebr72.p06	pdee118.p06
eeldx040.p06	eeldx085.p06	pdebr28.p06	pdebr73.p06	pdee119.p06
eeldx041.p06	eeldx086.p06	pdebr29.p06	pdebr74.p06	pdee120.p06
eeldx042.p06	eeldx087.p06	pdebr30.p06	pdebr75.p06	pdee121.p06
eeldx043.p06	eeldx088.p06	pdebr31.p06	pdebr76.p06	pdee122.p06
eeldx044.p06	eeldx089.p06	pdebr32.p06	pdebr77.p06	pdee123.p06
eeldx045.p06	eeldx090.p06	pdebr33.p06	pdebr78.p06	pdee124.p06

pdeel25.p06	pdeel70.p06	pdesi15.p06	pdesi60.p06	pdgph06.p06
pdeel26.p06	pdeel71.p06	pdesi16.p06	pdesi61.p06	pdgph07.p06
pdeel27.p06	pdeel72.p06	pdesi17.p06	pdesi62.p06	pdgph08.p06
pdeel28.p06	pdeel73.p06	pdesi18.p06	pdesi63.p06	pdgph09.p06
pdeel29.p06	pdeel74.p06	pdesi19.p06	pdesi64.p06	pdgph10.p06
pdeel30.p06	pdeel75.p06	pdesi20.p06	pdesi65.p06	pdgph11.p06
pdeel31.p06	pdeel76.p06	pdesi21.p06	pdesi66.p06	pdgph12.p06
pdeel32.p06	pdeel77.p06	pdesi22.p06	pdesi67.p06	pdgph13.p06
pdeel33.p06	pdeel78.p06	pdesi23.p06	pdesi68.p06	pdgph14.p06
pdeel34.p06	pdeel79.p06	pdesi24.p06	pdesi69.p06	pdgph15.p06
pdeel35.p06	pdeel80.p06	pdesi25.p06	pdesi70.p06	pdgph16.p06
pdeel36.p06	pdeel81.p06	pdesi26.p06	pdesi71.p06	pdgph17.p06
pdeel37.p06	pdeel82.p06	pdesi27.p06	pdesi72.p06	pdgph18.p06
pdeel38.p06	pdeel83.p06	pdesi28.p06	pdesi73.p06	pdgph19.p06
pdeel39.p06	pdeel84.p06	pdesi29.p06	pdesi74.p06	pdgph20.p06
pdeel40.p06	pdeel85.p06	pdesi30.p06	pdesi75.p06	pdgph21.p06
pdeel41.p06	pdeel86.p06	pdesi31.p06	pdesi76.p06	pdgph22.p06
pdeel42.p06	pdeel87.p06	pdesi32.p06	pdesi77.p06	pdgph23.p06
pdeel43.p06	pdeel88.p06	pdesi33.p06	pdesi78.p06	pdgph24.p06
pdeel44.p06	pdeel89.p06	pdesi34.p06	pdesi79.p06	pdgph25.p06
pdeel45.p06	pdeel90.p06	pdesi35.p06	pdesi80.p06	pdgph26.p06
pdeel46.p06	pdeel91.p06	pdesi36.p06	pdesi81.p06	pdgph27.p06
pdeel47.p06	pdeel92.p06	pdesi37.p06	pdesi82.p06	pdgph28.p06
pdeel48.p06	pdeel93.p06	pdesi38.p06	pdesi83.p06	pdgph29.p06
pdeel49.p06	pdeel94.p06	pdesi39.p06	pdesi84.p06	pdgph30.p06
pdeel50.p06	pdeel95.p06	pdesi40.p06	pdesi85.p06	pdgph31.p06
pdeel51.p06	pdeel96.p06	pdesi41.p06	pdesi86.p06	pdgph32.p06
pdeel52.p06	pdeel97.p06	pdesi42.p06	pdesi87.p06	pdgph33.p06
pdeel53.p06	pdeel98.p06	pdesi43.p06	pdesi88.p06	pdgph34.p06
pdeel54.p06	pdeel99.p06	pdesi44.p06	pdesi89.p06	pdgph35.p06
pdeel55.p06	pdeflist.p06	pdesi45.p06	pdesi90.p06	pdgph36.p06
pdeel56.p06	pdesi01.p06	pdesi46.p06	pdesi91.p06	pdgph37.p06
pdeel57.p06	pdesi02.p06	pdesi47.p06	pdesi92.p06	pdgph38.p06
pdeel58.p06	pdesi03.p06	pdesi48.p06	pdesi93.p06	pdgph39.p06
pdeel59.p06	pdesi04.p06	pdesi49.p06	pdesi94.p06	pdgph40.p06
pdeel60.p06	pdesi05.p06	pdesi50.p06	pdesi95.p06	pdgph41.p06
pdeel61.p06	pdesi06.p06	pdesi51.p06	pdesi96.p06	pdgph42.p06
pdeel62.p06	pdesi07.p06	pdesi52.p06	pdesi97.p06	pdgph43.p06
pdeel63.p06	pdesi08.p06	pdesi53.p06	pdesi98.p06	pdgph44.p06
pdeel64.p06	pdesi09.p06	pdesi54.p06	pdesi99.p06	pdgph45.p06
pdeel65.p06	pdesi10.p06	pdesi55.p06	pdgph01.p06	pdgph46.p06
pdeel66.p06	pdesi11.p06	pdesi56.p06	pdgph02.p06	pdgph47.p06
pdeel67.p06	pdesi12.p06	pdesi57.p06	pdgph03.p06	pdgph48.p06
pdeel68.p06	pdesi13.p06	pdesi58.p06	pdgph04.p06	pdgph49.p06
pdeel69.p06	pdesi14.p06	pdesi59.p06	pdgph05.p06	pdgph50.p06

pdgph51.p06 pdgph96.p06 pdgpp42.p06 pdgpp87.p06 pdpsi33.p06  
pdgph52.p06 pdgph97.p06 pdgpp43.p06 pdgpp88.p06 pdpsi34.p06  
pdgph53.p06 pdgph98.p06 pdgpp44.p06 pdgpp89.p06 pdpsi35.p06  
pdgph54.p06 pdgph99.p06 pdgpp45.p06 pdgpp90.p06 pdpsi36.p06  
pdgph55.p06 pdgpp01.p06 pdgpp46.p06 pdgpp91.p06 pdpsi37.p06  
pdgph56.p06 pdgpp02.p06 pdgpp47.p06 pdgpp92.p06 pdpsi38.p06  
pdgph57.p06 pdgpp03.p06 pdgpp48.p06 pdgpp93.p06 pdpsi39.p06  
pdgph58.p06 pdgpp04.p06 pdgpp49.p06 pdgpp94.p06 pdpsi40.p06  
pdgph59.p06 pdgpp05.p06 pdgpp50.p06 pdgpp95.p06 pdpsi41.p06  
pdgph60.p06 pdgpp06.p06 pdgpp51.p06 pdgpp96.p06 pdpsi42.p06  
pdgph61.p06 pdgpp07.p06 pdgpp52.p06 pdgpp97.p06 pdpsi43.p06  
pdgph62.p06 pdgpp08.p06 pdgpp53.p06 pdgpp98.p06 pdpsi44.p06  
pdgph63.p06 pdgpp09.p06 pdgpp54.p06 pdgpp99.p06 pdpsi45.p06  
pdgph64.p06 pdgpp10.p06 pdgpp55.p06 pdpsi01.p06 pdpsi46.p06  
pdgph65.p06 pdgpp11.p06 pdgpp56.p06 pdpsi02.p06 pdpsi47.p06  
pdgph66.p06 pdgpp12.p06 pdgpp57.p06 pdpsi03.p06 pdpsi48.p06  
pdgph67.p06 pdgpp13.p06 pdgpp58.p06 pdpsi04.p06 pdpsi49.p06  
pdgph68.p06 pdgpp14.p06 pdgpp59.p06 pdpsi05.p06 pdpsi50.p06  
pdgph69.p06 pdgpp15.p06 pdgpp60.p06 pdpsi06.p06 pdpsi51.p06  
pdgph70.p06 pdgpp16.p06 pdgpp61.p06 pdpsi07.p06 pdpsi52.p06  
pdgph71.p06 pdgpp17.p06 pdgpp62.p06 pdpsi08.p06 pdpsi53.p06  
pdgph72.p06 pdgpp18.p06 pdgpp63.p06 pdpsi09.p06 pdpsi54.p06  
pdgph73.p06 pdgpp19.p06 pdgpp64.p06 pdpsi10.p06 pdpsi55.p06  
pdgph74.p06 pdgpp20.p06 pdgpp65.p06 pdpsi11.p06 pdpsi56.p06  
pdgph75.p06 pdgpp21.p06 pdgpp66.p06 pdpsi12.p06 pdpsi57.p06  
pdgph76.p06 pdgpp22.p06 pdgpp67.p06 pdpsi13.p06 pdpsi58.p06  
pdgph77.p06 pdgpp23.p06 pdgpp68.p06 pdpsi14.p06 pdpsi59.p06  
pdgph78.p06 pdgpp24.p06 pdgpp69.p06 pdpsi15.p06 pdpsi60.p06  
pdgph79.p06 pdgpp25.p06 pdgpp70.p06 pdpsi16.p06 pdpsi61.p06  
pdgph80.p06 pdgpp26.p06 pdgpp71.p06 pdpsi17.p06 pdpsi62.p06  
pdgph81.p06 pdgpp27.p06 pdgpp72.p06 pdpsi18.p06 pdpsi63.p06  
pdgph82.p06 pdgpp28.p06 pdgpp73.p06 pdpsi19.p06 pdpsi64.p06  
pdgph83.p06 pdgpp29.p06 pdgpp74.p06 pdpsi20.p06 pdpsi65.p06  
pdgph84.p06 pdgpp30.p06 pdgpp75.p06 pdpsi21.p06 pdpsi66.p06  
pdgph85.p06 pdgpp31.p06 pdgpp76.p06 pdpsi22.p06 pdpsi67.p06  
pdgph86.p06 pdgpp32.p06 pdgpp77.p06 pdpsi23.p06 pdpsi68.p06  
pdgph87.p06 pdgpp33.p06 pdgpp78.p06 pdpsi24.p06 pdpsi69.p06  
pdgph88.p06 pdgpp34.p06 pdgpp79.p06 pdpsi25.p06 pdpsi70.p06  
pdgph89.p06 pdgpp35.p06 pdgpp80.p06 pdpsi26.p06 pdpsi71.p06  
pdgph90.p06 pdgpp36.p06 pdgpp81.p06 pdpsi27.p06 pdpsi72.p06  
pdgph91.p06 pdgpp37.p06 pdgpp82.p06 pdpsi28.p06 pdpsi73.p06  
pdgph92.p06 pdgpp38.p06 pdgpp83.p06 pdpsi29.p06 pdpsi74.p06  
pdgph93.p06 pdgpp39.p06 pdgpp84.p06 pdpsi30.p06 pdpsi75.p06  
pdgph94.p06 pdgpp40.p06 pdgpp85.p06 pdpsi31.p06 pdpsi76.p06  
pdgph95.p06 pdgpp41.p06 pdgpp86.p06 pdpsi32.p06 pdpsi77.p06

pdpsi78.p06 pelDX023.p06 pelDX068.p06  
pdpsi79.p06 pelDX024.p06 pelDX069.p06  
pdpsi80.p06 pelDX025.p06 pelDX070.p06  
pdpsi81.p06 pelDX026.p06 pelDX071.p06  
pdpsi82.p06 pelDX027.p06 pelDX072.p06  
pdpsi83.p06 pelDX028.p06 pelDX073.p06  
pdpsi84.p06 pelDX029.p06 pelDX074.p06  
pdpsi85.p06 pelDX030.p06 pelDX075.p06  
pdpsi86.p06 pelDX031.p06 pelDX076.p06  
pdpsi87.p06 pelDX032.p06 pelDX077.p06  
pdpsi88.p06 pelDX033.p06 pelDX078.p06  
pdpsi89.p06 pelDX034.p06 pelDX079.p06  
pdpsi90.p06 pelDX035.p06 pelDX080.p06  
pdpsi91.p06 pelDX036.p06 pelDX081.p06  
pdpsi92.p06 pelDX037.p06 pelDX082.p06  
pdpsi93.p06 pelDX038.p06 pelDX083.p06  
pdpsi94.p06 pelDX039.p06 pelDX084.p06  
pdpsi95.p06 pelDX040.p06 pelDX085.p06  
pdpsi96.p06 pelDX041.p06 pelDX086.p06  
pdpsi97.p06 pelDX042.p06 pelDX087.p06  
pdpsi98.p06 pelDX043.p06 pelDX088.p06  
pdpsi99.p06 pelDX044.p06 pelDX089.p06  
pdrelax.p06 pelDX045.p06 pelDX090.p06  
pelDX001.p06 pelDX046.p06 pelDX091.p06  
pelDX002.p06 pelDX047.p06 pelDX092.p06  
pelDX003.p06 pelDX048.p06 pelDX093.p06  
pelDX004.p06 pelDX049.p06 pelDX094.p06  
pelDX005.p06 pelDX050.p06 pelDX095.p06  
pelDX006.p06 pelDX051.p06 pelDX096.p06  
pelDX007.p06 pelDX052.p06 pelDX097.p06  
pelDX008.p06 pelDX053.p06 pelDX098.p06  
pelDX009.p06 pelDX054.p06 pelDX099.p06  
pelDX010.p06 pelDX055.p06  
pelDX011.p06 pelDX056.p06  
pelDX012.p06 pelDX057.p06  
pelDX013.p06 pelDX058.p06  
pelDX014.p06 pelDX059.p06  
pelDX015.p06 pelDX060.p06  
pelDX016.p06 pelDX061.p06  
pelDX017.p06 pelDX062.p06  
pelDX018.p06 pelDX063.p06  
pelDX019.p06 pelDX064.p06  
pelDX020.p06 pelDX065.p06  
pelDX021.p06 pelDX066.p06  
pelDX022.p06 pelDX067.p06

# Index

- absorption energy, choosing, [35](#)
- analysis script, [74](#)
  - Element, [75](#)
  - Input, [74](#)
  - Matrix, [75](#)
  - NScan, [75](#)
  - Output, [74](#)
  - Path, [74](#)
  - Point, [74](#)
  - Region, [75](#)
  - Scan, [74](#)
- bremsstrahlung, [36](#)
- calculation, stages, [41](#)
- command
  - MCA, [50](#)
- commands
  - BCalc, [28](#)
  - BFile2D, [24](#)
  - BFile3D, [24](#)
  - BTable, [24](#)
  - BUni, [17, 25](#)
  - C1, [35](#)
  - C2, [35](#)
  - DsMax, [21, 37](#)
  - DUnit, [16, 23](#)
  - EAbs, [20, 35](#)
  - ECalc, [28](#)
  - EFile2D, [23](#)
  - EFile3D, [24](#)
  - EMax, [20, 36](#)
  - Enhance, [37](#)
  - EscapeFilter, [40](#)
  - EscapeOff, [39](#)
  - Force, [20, 36](#)
  - FullInfo, [28](#)
  - GFile2D, [17, 23](#)
  - GFile3D, [23](#)
  - Material, [26](#)
  - MCA, [40](#)
  - NPMult, [19, 32](#)
  - NSearchWidth, [38](#)
  - Parallel, [39, 44](#)
  - PFile, [19, 32](#)
  - PList, [19, 31](#)
  - PlotOn, [39](#)
  - Reduce, [37](#)
  - Region, [27](#)
  - Rotate, [19, 24, 25, 32](#)
  - RSeed, [36](#)
  - SFile, [19](#)
  - Shift, [19, 24, 25, 33](#)
  - SList, [19, 30, 31](#)
  - StepLimit, [39](#)
  - StepMax, [21, 38](#)
  - Time, [21, 38](#)
  - TPulse, [19, 32](#)
  - WCC, [35](#)
  - WCR, [35](#)
- Composition section, [15, 18, 26](#)
- deposited energy file, [50](#)
- dose file, [48](#)
- dose units, [12, 32, 48, 53](#)
- DsMax, choosing value, [37](#)
- electric field, [22, 23](#)
- escape file, [28, 48](#)
- exclusion region, [28](#)
- GamBet
  - background operation, [42](#)
  - batch file control, [43](#)
  - file types, [48](#)
  - parallel operation, [44](#)
  - program menu, [41](#)
- GBLaunch, [8, 41](#)
  - Create and run tasks, [8](#)
- GBView2 commands
  - Close data listing, [55](#)
  - Close trajectory file, [55](#)
  - Contour plot type, [57](#)
  - Create script, [56](#)
  - Edit data file, [56](#)
  - Edit file, [56](#)
  - Edit script, [56](#)
  - Element outline, [57](#)
  - Element properties, [60](#)
  - Line scan, [59](#)
  - Load dose file, [55](#)
  - Load trajectory file, [55](#)
  - Matrix file, [60](#)

- Mouse/keyboard, [57](#)
- Open data listing, [55](#)
- Oscilloscope mode, [62](#)
- Plot export, [59](#)
- Plot limits, [57](#)
- Plot type, [56](#)
- Plotted quantity, [57](#)
- Point calculation, [59](#)
- Region properties, [60](#)
- Run script, [56](#)
- Save dose file, [61](#)
- Save plot file, [59](#)
- Scan quantity, [60](#)
- Smooth dose, [61](#)
- Snap mode, [57](#)
- Toggle grid, [57](#)
- Volume integrals, [60](#)
- GBView3 commands
  - Create script, [64](#)
  - Cut planes, [71](#)
  - Displayed regions, [71](#)
  - Dose file information, [63](#)
  - Edit, [64](#)
  - Line scan, [65](#), [70](#)
  - Load dose file, [63](#)
  - Load named view, [72](#)
  - Load trajectory file, [63](#)
  - Matrix file, [65](#)
  - Open data listing, [63](#)
  - Oscilloscope mode, [71](#)
  - Plot export, [67](#)
  - Plot quantity, [67](#)
  - Plot style, [67](#), [69](#), [71](#)
  - Point calculation, [64](#), [70](#)
  - Region properties, [65](#)
  - Run script, [64](#)
  - Save dose file, [66](#)
  - Save named view, [72](#)
  - Save plot file, [67](#)
  - Set plane, [67](#)
  - Slice plane, [68](#)
  - Smooth dose, [66](#)
  - Volume integrals, [65](#)
- GBView3 plot style
  - plane, [66](#)
  - slice, [68](#)
  - surface, [71](#)
- geometry mesh, [16](#), [22](#)
- Geometry section, [15](#), [16](#), [22](#)
- installation instructions, [7](#)
- interaction forcing, [20](#), [36](#)
- Listing file, [48](#), [51](#)
- magnetic field, [22](#), [24](#)
  - beam-generated, [24](#)
  - table format, [24](#)
- material
  - defining new, [27](#)
- meshes, rules for combining, [22](#)
- multi-channel analyzer, [40](#), [50](#)
- OmniTrak, [5](#)
  - PRT file, [18](#), [29](#), [31](#)
- package components, [5](#)
- pair production, [36](#)
- parallel operation sequence, [44](#)
- parallel processing, [7](#)
  - child process, [44](#)
  - combining data, [45](#)
  - distributed memory architecture, [44](#)
  - file conflict, [45](#)
  - listing file, [45](#)
  - operation sequence, [44](#)
  - primary process, [44](#)
  - run time reduction, [46](#)
  - shared memory architecture, [44](#)
- particle
  - flux units, [12](#), [49](#), [54](#)
  - primary, definition, [29](#)
  - properties, [30](#), [31](#)
- Penelope, [5](#), [7](#), [15](#), [19](#), [28](#), [37](#)
  - data files, [82](#)
  - manual, [34](#)
  - materials, [18](#), [26](#), [76](#)
  - setup information, [51](#)
- Plot views, saving, [72](#)
- Process section, [15](#), [19](#), [34](#)
- program function, [5](#)
- random number seed, [36](#), [47](#)
- run statistics, [48](#)
- run time
  - optimizing, [36](#)
- script
  - example, [15](#)
  - Setup dialog, [16](#)

- structure, [14](#)
- shower, definition, [29](#), [32](#)
- simulation modes, [12](#)
- source information, generating, [29](#)
- Source section, [15](#), [18](#), [29](#)
- standard delimiters, [14](#)
- statistical quantities, [51](#)
  - dose, [53](#)
  - energy conservation, [53](#)
  - primary particles, [52](#)
- tasks
  - in GBLaunch, [8](#)
- trajectory file, [49](#)
- Trak, [5](#), [23](#)
  - PRT file, [18](#), [29](#), [31](#)
- variance reduction, [36](#)
- void region, [28](#)

Penelope is covered by the following copyright:

PENELOPE/PENGEOM (version 2006)

Copyright (c) 2001-2006

Universitat de Barcelona

Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in all supporting documentation. The Universitat de Barcelona makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.