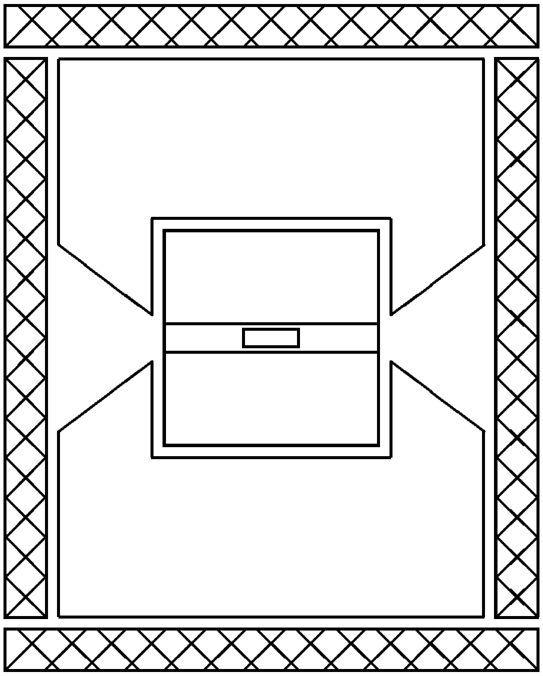
CB–Note 169/Version 1.1

LEAR Crystal Barrel Experiment, PS197
Monte Carlo Software
CBGEANT 4.06/03

Roy Bossingham
University of California, Berkeley

12 February, 1992



Contents

List of Tables

1 Introduction

CBGEANT is the basic event simulation package for the Crystal Barrel detector. It initiates an event, describes geometry, extracts data about energy deposition in sensitive detector elements, simulates detector response, applies experimental distortions and records output data in a format compatible with actual events to allow later analysis by the standard programs. The CERN code package GEANT, version 3.14 or 3.15 (the corresponding CBGEANT code being extracted with a version number patch), is used to propagate the initial particles along with their decay and/or shower products, and to apply energy loss, scattering and the dominant electromagnetic and hadronic processes to which they are subject. Like the well-known EGS4 Electron-Gamma Shower code, GEANT is a condensed history Monte Carlo. (While some of GEANT's algorithms have been borrowed from EGS4, the actual EGS4 code is not a subset of GEANT.)

Condensed history Monte Carlos tend to be CPU intensive, and CBGEANT is no exception. While there are wide variations depending on beam energy, decay channel and random statistics, an event typically requires 11 seconds on a SPARCstation I, 4.5 seconds on the CBDEC DECstations or 40 seconds on a VAX 3600. Since a serious Monte Carlo run usually involves upwards of a thousand events, jobs are usually run in batch mode, often in the background.

However, CBGEANT/GEANT also support an interactive mode in which the user may display various views of detector components, as well as produce and plot event tracks superimposed on predefined views. This allows the user to detect coding errors and to directly visualize detector effects, which is very useful (if not essential) for code debugging and development.

All CBGEANT code is written in Fortran. The machines which it currently supports are the Zurich Alliant under UNIX, DECstations under ULTRIX, CERNVM under IBM/CMS, SPARCstations and VAX/VMS. It is likely that the code will run on other UNIX machines with relatively few changes when extracted for a similar, supported machine. However, there are machine dependencies in I/O handling, and extentions to ANSI-77 Fortran occasionally appear, so one must expect to encounter *some* problems when transporting the code. Considerable effort could be required for some machines, especially if the CERN libraries are not supported on them.

The present CBGEANT incarnation builds on version 4.02, but contains many major changes; most results should *not* be consistent between them. In general, user code must be recompiled due to the redefined common blocks in CBGEANT and GEANT. A more detailed comparison appears in a later section, but some of the substantive changes to CBGEANT are:

- The geometry and materials sections have been rewritten with much greater detail and accuracy; in many respects, the detector in version 4.02 was the proposal detector, rather than the real device. An incompatibility in the output data arises from the rotations of the JDC and PWC to their approximate positions in the real detector.

- More physics processes are now implemented by default. Default energy thresholds for particle production and tracking have been lowered to improve the shower simulation accuracy. Improvements in GEANT 3.14 have made a number of other physics improvements possible, including the inclusion of Landau straggling and handling of nuclear fragmentation by the NUCRIN code. The intention is for CBGEANT to give a *reasonably* accurate, general-purpose detector simulation with the default settings.

- The CBGEANT particle definitions have been improved. Parameters were updated to 1990 Particle Data Book values, and branching to forbidden channels was eliminated.
- There is more flexibility in the choice of annihilation vertex distributions, and non-uniformities in the particle distributions from several of the test generators were fixed. Support for additional annihilation channels was added, and the branching ratios for unbiased decay were improved; a larger fraction of the branching ratio was put into intermediate states. In addition, LD_2 targets are now supported.
- Fixes for several severe GEANT tracking bugs involving type-'many' volumes have been provided for GEANT 3.14. These have been fixed in the version 3.15 release.
- Additional non-ideal detector effects are now supported, including dE/dx dependence in CsI light output, variation in crystal calibrations and separate random levels of coherent and incoherent noise for FERA and 2282 ADC's. The crystal calibration and noise can be individually set for crystals.
- Tracking parameters have been changed in many places, and non-helical tracking is now supported in the central part of the detector. The sign of the magnetic field has also been corrected, and fresh tables of JDC drift time data are provided.
- Neutrinos are no longer tracked, and other particles are not tracked beyond the point at which their time of flight exceeds the crystal ADC gate duration. On the other hand, particles are no longer rejected when they go further than 85 cm from the Z axis.
- A number of new data cards increases the flexibility of the code. Data cards will now be read from logical unit 99 (except on CERNVM), taking them from standard input if this fails.
- Exchange-mode files and tapes are now supported on UNIX systems.
- Additional data has been put into the informational data bank, allowing automatic setting of detector parameters in the analysis code and verification of the production conditions for a data set.
- Numerous bugs were fixed, a few new debugging options were added and the code is more compliant with ANSI-77 Fortran standards.
- The documentation and scripts in the `$HISTORY` patch have been rewritten and extended.

The primary cost of the new version is the increased CPU time per event: with the default parameters for both codes, CBGEANT 4.06/03 runs half as fast as version 4.02/10. Almost all of this cost is due to particles having lower production thresholds/cuts and being tracked into the magnet coil and beyond; the increased complexity of the geometry has little impact (at least on a computer with enough memory not to page fault). Actually, it is GEANT, rather than CBGEANT, in which the bulk of the CPU time is spent, since GEANT goes through rather painful contortions to determine geometrical regions or to do anything else). One cannot expect much direct gain from streamlining the CBGEANT code, and, since the CPU time is *not* highly sensitive to the particle production thresholds and cuts, one cannot expect much improvement by fine-tuning these parameters. Some improvement might come from optimizing the volume searching, but many of the obvious improvements have already been made.

The CPU time required for run initialization is also increased (typically a minute), which can be annoying when running interactively, but is negligible for a serious job. The setup time can be cut significantly by using the `CCUT` data card when reducing the setup time is more important than the execution time (e.g., for debugging).

Additional demands are also made on memory by the new code. In particular, problems may arise with interactive runs at high beam energies. It may sometimes be necessary to increase the memory allocation in the main routine of CBGEANT. For interactive running, memory is allocated by a GEANT substitute main program GXINT, so that one may need to change GXINT. This is done by selecting the `BIG` or `BIGBIG` patch when extracting GXINT.

The code has benefited from pre-release testing on various machines by Tony Noble, Fritz-Herbert Hensius, Curtis Meyer, Gunter Folger, Hans-Peter Dietz and Klaus Peters. Also, Nigel Hessey made a number of useful suggestions. Several people besides myself have contributed directly to coding the present version of CBGEANT, including Joachim Zaers (who spent some weeks coding parts of the geometry); Christoph Strassburger (who did part of the updating of the UGPART subroutine and provided the LD_2 annihilation code); and Stephen Spanier and Klaus Peters (who have modified the JDC signal routines somewhat).

Users will undoubtedly find residual bugs; comments on these or other matters can be addressed to:

- Michael Kobel (overall responsibility, 1992) KOBEL@CERNVM.BITNET
- Tony Noble (responsible for crystal coding): POTATO@CERNVM.BITNET
- Klaus Peters (responsible for JDC coding): PETERS@VIPMZA.PHYSIK.UNI-MAINZ.DE
- Roy Bossingham: (versions 4.03–4.06) RRBOSSINGHAM@LBL.GOV

CB Note #103 (*The Monte Carlo Programmer's Guide*) by Klaus Peters may be useful to the CBGEANT user, particularly its description of many subroutines. However, the present version is no longer complete or entirely accurate, as it precedes this code release by two or three years. Other information may be available from earlier CBGEANT authors: Kersten Braune, Gunter Folger, Fritz-Herbert Heinsius, Marcel Kunze, Klaus Peters and Curtis

Meyer. M. Guckes and B. Lewendel are no longer with the collaboration. (Gunter Folger is now in the DD Division.)

2 Installation

2.1 GEANT Support

The first thing to verify is that one has a current version of GEANT. Testing was mainly done with version 3.14/16, correction cradle 3.14/07. Versions before 3.14 are not supported and will not work, while results with correction cradles before 3.14/07 cannot be guaranteed. Some earlier versions will give fatal errors on some machines, while others will quietly yield incorrect results.

CBGEANT attempts to support GEANT 3.15, and some testing has been done with prerelease versions. These give similar results and run at nearly the same speed as 3.14 on a SPARCstation 1, but, interestingly, ran at twice the speed of 3.14 on the Alliant vector processor in tests by Curtis Meyer. GEANT 3.15 has evolved significantly during its prerelease testing phase, and earlier versions may not perform correctly due to changes in common blocks, bug fixes, lack of support for type 'many' volumes, etc. At a minimum, verify that the version is dated 10-FEB-1992 or later. As of this writing, one may obtain the relavant files with anonymous ftp from CERNVM:

```
$ FTP 128.141.2.4
  anonymous
ftp> CD LIBRARY.221
ftp> GET GEANG315.CAR
ftp> GET GEANH315.CAR
ftp> GET GEANT315.CAR
ftp> GET GEANX315.CAR
ftp> GET GEANT315.CRA
ftp> GET GEANX315.CRA
ftp> GET GXINT315.CRA
```

The GEANT code is still being repaired and updated at CERN. New bugs, many of them significant, are reported (and often fixed) on a regular basis. The user should, therefore, use an up-to-date release of GEANT, and avoid using the code blindly. It can be very enlightening to dump a few events step-by-step with the `DMPSTP` patch. If the code's behavior does not seem to make sense, there is a good chance that you have found another bug (or made a linking error or ...).

The default memory allocations in GEANT's main routine for the interactive version, GXINT, are marginal and should be increased by selecting `BIG` when producing the code or by setting ones own values in the code. Reasonable starting values are: `NWGEAN=500000` and `NWPAW=200000`. If these are not enough, ZEBRA will crash in an inelegant and uniformative way; try increasing them further. Memory requirements increase for higher particle energies, lower particle production and tracking thresholds, for more complex geometries and with the extraction of additional data from events.

The person in charge of maintaining GEANT at CERN is:

```
Federico Carminati
CERN-CN
1211 Geneva 23
41-22-767-4959
FCA@CERNVM.BITNET or VXCERN::FCA
```

One should obtain the GEANT documentation from CERN, since a number of data cards and parameters are described in it and it is needed to understand much of CBGEANT. As of now, the GEANT documentation consists of a 3.15 draft manual (rough, with mistakes, but slated for an early 1992 release); the 1987 manual (obsolete and no longer distributed); the `HISTORY` patch of the GEANT code; and occasional articles in the CERN Computer Newsletter. Problems with GEANT and its future development are discussed at monthly GEANT user meetings held at CERN. The dates for these meetings, and discussions related to GEANT problems and development are available through the electronic discussion group. One may subscribe by sending a message to LISTSERV@CERNVM with a single-line body:

```
SUBSCRIBE LGEANT Firstname Lastname Institute
```

To get the most recent history patches and correction cradles, send mail to LISTSERV@CERNVM with the single lines:

```
GET GCORR HISTORY
```

or

```
GET GCORR CRADLE
```

2.2 Loading CBGEANT into CMZ

The CBGEANT code is maintained under CMZ, but is usually distributed as a card file. One should reload this into CMZ using a current CMZ version, since versions before mid-1991 are unable to handle the large data decks in the `DFILES` patch. (If one is forced to use an obsolete version of CMZ, one can strip the `DFILES` patch from the card file before proceeding. An editor such as EMACS, which does not balk at large files, is recommended.) To load the card file named `CBGEANT040601.CAR`, start CMZ, then:

```
    CMZ> SEL .
    CMZ> SEQ .
    CMZ> PILOT .
    CMZ> RELEASE *
    CMZ> MAKE cbgeant
    CMZ> YTOC cbgeant040601.car
```

2.3 Patch Descriptions

The CBGEANT patches are of several types. The informational ones are:

$VERSION	Descriptions of recent versions of CBGEANT.
$HISTORY	Instruction, documentation, .kumac files and UNIX scripts

Machine patches are:

ALT	Zurich Alliant, running UNIX
DECS	DECstations, running ULTRIX
IBM	CERNVM, running CMS
SUN	SPARCstations, running UNIX
VAX	VAXen, running VMS

The pilot patches are listed next. Those corresponding to the above machines are for getting the standard batch version of CBGEANT, excepting the main, dummy user and GEANT override routines. All of these use `FASTJDC` and `*BATCH`. The user must externally select the version of GEANT to get the correct common blocks, specifying `GEA314` or `GEA315`.

***ALLIANT**	Zurich Alliant
***DECS**	DECstations (MIPS compiler)
***IBM**	CERNVM
***SUN**	SPARCstations
***VAX**	VAX
***CBUSER**	Gets main, user and override routines only
***BATCHF**	Standard batch version with fast routines
***BATCH**	Standard batch version
***INTERF**	Interactive/graphics version with fast routines
***INTER**	Interactive/graphics version

The patches holding the Fortran decks are:

GCDES	GEANT and ZEBRA common blocks
COMMCB	CBGEANT common blocks
CBBASE	Base routines of CBGEANT
CBDRIF	Drift chamber utility routines
CBERRM	Error and warning message handlers
CBIO	Several I/O routines for drift chamber files
CBKINE	Routines for initializing events
CBMATE	Material and tracking media definitions
CBPROC	Process control routines
CBUSER	Dummy user routines
CBVIEW	Interactive graphics and plot definitions
BC	Crystal geometry and digitisation routines
GH	GHT geometry and digitisation routines
JD	JDC geometry and digitisation routines
MG	Magnet and magnetic field routines
PW	PWC geometry and digitisation routines
SF	SFD geometry and digitisation routines (not implemented)
VE	Veto counter routines
OBSOLETE	Code no longer needed for linking: subject to deletion

There are several modes which select code for inclusion:

NEVER	Marks lines not to be compiled
DMPSTP	Dumps information in GUSTEP for each step
DMPVOL	Dumps volume search tree in GUSTEP for each step
FASTJDC	Switches to fast JDC processing
FASTXTAL	Switches to fast crystal processing (not implemented)
SUNERR	Primitive Sun IEEE floating point error handler

2.4 Documentation and Auxillary Files

Before producing the code itelf, there are a number of auxillary files which one may wish to extract. First, there are documentation files in the `$HISTORY` patch. Together, these occupy ~100,000 bytes (200 VAX blocks).

cards.doc	Data card descriptions
channel.doc	$\overline{p}p$ annihilation channels supported
ecut.doc	Discussion of particle production thresholds
geantinf.doc	Information on GEANT
history.doc	CBGEANT history
instruct.doc	Instructions for installing and using CBGEANT
material.doc	Explanation of many CBGEANT material definitions
media.doc	List of materials and tracking media used in CBGEANT
particle.doc	List of particles defined for GEANT and CBGEANT
physics.doc	Physics deficiencies in GEANT—a partial list
re_unix.doc	Setup and use of GEANT under UNIX
volumes.doc	Nesting and tracking media of the declared volumes

Next, the `DFILES` patch holds the current JDC drift time tables which have been calculated for the three magnetic field settings commonly used by the Crystal Barrel. Each of these formatted files occupies 450,590 bytes (890 VAX blocks).

jdc00.dat	Zero field drift table (JDC calibration)
jdc10.dat	10 kilogauss drift table (crystal calibration)
jdc15.dat	15 kilogauss drift table (normal data taking)

There is a simple procedure to extract all documentation and JDC drift time files:

```
CMZ> FILE cbgeant -R
CMZ> SET  datdoc.kumac -D
CMZ> CD   $HISTORY
CMZ> CTOT DATDOCKU
CMZ> EXEC datdoc
CMZ> EXIT
```

If disk space is scarce, one may prefer to extract individual files. For example, to get the deck `INSTRUCT`:

```
CMZ> FILE cbgeant -R
CMZ> SET  instruct.doc -D
CMZ> CD   $HISTORY
CMZ> CTOT INSTRUCT
CMZ> EXIT
```

Similarly, one may extract individual drift time files, such as `jdc15.dat`:

```
CMZ> FILE cbgeant -R
CMZ> SET  jdc15.dat -D
CMZ> CD   DFILES
CMZ> CTOT JDC15
CMZ> EXIT
```

2.5 Extracting CBGEANT

One may use the KUMAC cradles supplied in the `$HISTORY` patch to extract the code. Start CMZ, clearing any existing selections, sequences and pilots:

```
CMZ> SEL .
CMZ> SEQ .
CMZ> PILOT .
```

To get a graphics/interactive version of the code, next use the line:

```
CMZ> SELECT INTERF
```

In all cases, use:

```
CMZ> SELECT SUN      (substitute ALT, DECS, IBM or VAX, as appropriate)
CMZ> SELECT GEA314   (substitute GEA315, if appropriate)
CMZ> FILE cbgeant -R
CMZ> SET  cbgeant.kumac -D
CMZ> CD   $HISTORY
CMZ> CTOT -Y CBKUMAC
CMZ> SET  cbuser.kumac -D
CMZ> CTOT -Y USKUMAC
CMZ> EXIT
```

The `CBGEANT.KUMAC` file should then hold a pilot line for the chosen machine (e.g., `PILOT *SUN`), a line for setting the location of output source files (e.g., `SET src/*.f -F`) and a line to select common blocks for the correct GEANT version (e.g. `SELECT GEA314`). If your machine is not supported, choose the machine closest in type and try it; this may or may not work. You may, in this case, have to edit the `SET...-F` line. Remember to create the subdirectory `src`, `[.SRC]`, etc. as required. If you want to extract all currently used Fortran (subject to the usual selections), the `FORTALL.KUMAC` file is designed for this. (It does not give exactly the same thing as `cmz> ctof *`, because it ignores the `OBSOLETE` patch of code destined for deletion.)

```
CMZ> SELECT SUN      (substitute ALT, DECS, IBM or VAX, as appropriate)
CMZ> SELECT GEA314   (substitute GEA315, if appropriate)
CMZ> FILE cbgeant -R
CMZ> SET fortall.kumac -D
CMZ> CD $HISTORY
CMZ> CTOT -Y ALLKUMAC
CMZ> EXIT
```

2.5.1 Sun and Alliant Machines

If your machine is a Sun or an ALlianT, there is a shell script to simplify the rest of the procedure. (It is probably not hard to modify the Sun script for the DECstation.) Extract the script with the procedure below, then edit the installation-dependent aspects of the `cbgeant_mak` script and run it.

```
CMZ> FILE cbgeant -R
CMZ> CD $HISTORY
CMZ> SELECT SUN  (or ALT, as appropriate)
CMZ> SET cbgeant_mak -D
CMZ> CTOT -Y SHELL
CMZ> EXIT
```

2.5.2 CERNVM, VAX and DECStations

For other machines, or if you do not want to use the shell script, do the following:

```
CMZ> EXEC CBGEANT
CMZ> EXIT
```

You will now have the current CBGEANT Fortran source code, with the exception of the main routine, user routines and GEANT override routines. Extract the remaining useful code:

```
CMZ> EXEC CBUSER
CMZ> EXIT
```

This will place the code into `cbgeant.f`, `cbgeant.for`, etc. as appropriate for your machine. One may edit the user routines in this file as needed.

2.6 Compiling CBGEANT

It is probably best to compile the main routine, user routines and GEANT override routines separate from each other and from the remaining CBGEANT routines, which are often placed into a library. If one is using a machine (such as a VAX) on which it is convenient to force module extraction from a library and on which the library and file system can handle large blocks of modules, it is possible to compile all modules together. On UNIX machines one will probably want to separately compile the routines to be placed in a library. One may need to specify compiler options:

- Bounds checking on arrays is *not* possible because of the equivalencing of arrays with incompatible lengths in GEANT, however appealing this would be for debugging. The root of the problem is that memory for ZEBRA is allocated at *link* time, based on the size of a common block in the main routine, while bounds checking is established at compile time.
- One must not strip trailing blanks from the code or use a procedure which does, since Hollerith strings may be continued across lines. (CERN's “fprep” routine causes problems, for example.) Similarly, one should not pad the lines with blanks.
- One must not compile past column 72, since this has been used as a comment field in some routines.

- One may need to increase the allowed number of continuation lines; the usual 19 lines is not enough. Set it to 40 lines, or so.
- It should *not* be necessary to tell the compiler to save variables and common blocks between subroutine calls. All common blocks and (hopefully) all variables holding data between calls to a subroutine have been specified to be saved. However, excepting possible costs in CPU time and memory, it should do no harm to save all variables. On machines with adequate RAM memory, this may even increase the speed.

One may get warning messages, depending on the compiler type and the warning level set. For instance, although CBGEANT variables are initialized before use, this is sometimes done in ways not apparent to the compiler and warnings may be printed. There may also be warnings about non-ANSI standard code, including Hollerith constants, character data being assigned to non-character variables, list-directed formatting on an internal file and incompatible common block lengths. Usually, these warnings can be ignored.

In addition, UNIX machines tend to suffer from Fortran compiler bugs. Some of these problems arise from the compiler using the C-language library instead of an actual Fortran library. (Fortran support under UNIX is often given in the same spirit and with as much grace as a Rolls Royce dealer selling a used Volkswagen bug to a leperous peasant). A couple of constructions (e.g, the `INQUIRE` keyword) known to be problematic have been avoided.

2.7 Linking CBGEANT

Linking the code requires the forced inclusion of a main routine (either the CBGEANT routine GCB for the batch version or the CERN routine GXINT for the interactive version). Until the distribution GEANT version has been repaired, one must also force the inclusion of the GEANT override routines. One will also often want to force the linking of user routines in preference to the dummy ones. Finally, link with the Crystal Barrel BCTRAK library, with the CERN GEANT, PACKLIB and KERNLIB libraries and with a graphics library (often GKS).

The GEANT library should be a recent update of version 3.14 (or, for pioneers, 3.15). On UNIX machines, the ZEBRA version (probably included in the PACKLIB library) should be recent enough to support C-library data transfer—at least version 3.66, released in February 1991—and KERNLIB must include the CF package. (Look for the routine CFOPEN if in doubt.) For GEANT3.15, HIGZ must be at least version 1.13/09 (04-FEB-1992).

2.8 Compile/Link Examples

As a concrete example, the below VAX/VMS command file has been used on VSXTAL. It assumes that CBGEANT.FOR holds GCB, MYINIT, MYEVNT, MYLAST, GFLTHE, GINPCO, GINPGO, GINVOL, GMEDIA and GTMEDI (as it would after execution of CBGEANT.KUMAC) and that [.SRC]*.FOR contains all current CBGEANT code except for the main, user and GEANT override routines (as it would after execution of CBUSER.KUMAC). Naturally, the definition of various logicals and library names and locations will vary. The number of workable alternatives is nearly infinite, depending on the needs of the user, and there is no attempt here to be comprehensive.

```
$! CBGEANT.BLD
$!
$ COPY [.SRC]*.FOR CBGEANT_MOST.FOR
$ FORTRAN/NOEXTEND CBGEANT_MOST
$ FORTRAN/NOEXTEND CBGEANT
$ LIBRARY/CREATE CBGEANT_MOST CBGEANT_MOST
$ DELETE CBGEANT_MOST.OBJ;*
$!
  LINK/EXE=CBGEANT -
  CBGEANT.OBJ, -
  CBGEANT_MOST/LIB, -
  [CBO.CBOFF]BCTRAK.OLB/LIB, -
  GEANT.OLB/LIB, PACKLIB.OLB/LIB/INCL=QNEXT,
  KERNLIB.OLB/LIB/INCL=(FLOARG,VMAX,VMIN), -
  GRAFGKS.OLB/LIB, GKSKERNEL.OLB/LIB, GKSDUM.OLB/LIB, GKSDRIV.OLB/LIB
$!
$ DELETE CBGEANT.OBJ;*
$ PURGE CBGEANT.EXE
$ PURGE CBGEANT_MOST.FOR
$!
$! End of file
```

3 Comparison: CBGEANT 4.06/03 vs. 4.02/10

3.1 Geometry

There have been hundreds of changes in the detector geometry. Most of these do not need discussing, but some will be readily apparent:

- The liquid hydrogen target was formerly treated as a Mylar cylinder of 4 cm diameter and 40 cm length. In reality, the hydrogen is confined to a cylinder of 1.7 cm diameter and ~4 cm length, with an aluminum vacuum tube around it. There is also a stainless steel ring at one end (with some copper added in 1991) which can cause a significant number of γ conversions. Fillings of either LH_2 or LD_2 are supported, although the correct annihilation physics must be set separately.
- The beamline was not included previously; one would expect, on the average, that it would absorb energy from showers near the Z axis, reducing the calorimeter resolution.
- The veto counter was formerly ignored. Because of its proximity to the target, and the fact that is an active veto, it may cause distortions. It is now included, although it is up to the user to position it with a data card. Energy deposition in it is not yet recorded in a data bank.
- The Rohacell/Mylar walls of the PWC were formerly defined with a density which was an order of magnitude too high due to a mistake in averaging the density of the two components; now fixed.
- The Mainz gaseous hydrogen target (GHT) geometry is now coded, although there is not yet any digitization for it.
- The JDC description was formerly rather simple. Compared to the actual device, the old code had an inner carbon cylinder and endplates that were too thin. A substantial amount of structure is included on the chamber now, such as discrete preamps, cooling plates and the cable mass.
- The JDC is now alligned with the barrel calorimeter as it is in reality, a rotation of 84.97° from the old configuration. This and the PWC or GHT rotation can be adjusted by the user.
- In the barrel calorimeter, the crystals themselves were represented fairly accurately in the old code, except for some problems with type #13. However, the cornets around the crystals had an average density that was too low, exagerating the energy resolution. Materials radially beyond the crystals was also given an overly low density, possibly affecting split-off events associated with particle leakage through the crystals.
- The sign of the magnetic field is set to point against the beam direction.
- The magnet coil was formerly given an inner radius of 84.5 cm, rather than 76.5 cm; again, this might affect split-off events.

- The mother volumes of several detectors have been changed and defined to hold a tracking medium with higher tracking accuracy than the crude, default air that pervades the detector. This evades the problem that GEANT usually does not detect volumes with dimensions less than the tracking precision—1 cm in the case of the old default air. As an example, particles usually failed to see the inner wall of the PWC before.

- Runge-Kutta (non-helix) tracking is now an option for the user interested in detailed studies of the JDC tracking. The inhomogeneity of the magnetic field is small, but significant for some tracks when one is studying 100 μm tracking accuracies.

3.2 Default Physics and Cuts

Most of the CBGEANT 4.02/10 particle production energy thresholds were simply the GEANT defaults. This is not a problem as long as the user realizes that Monte Carlo defaults are usually written with the LEP detectors in mind (where no one cares about a few MeV one way or the other) and overrides them appropriately. Unfortunately, the hapless user is usually busy, wants to get results for the next meeting and does not want to spend days trying to understand GEANT from an outdated manual. The end result is that many old Monte Carlo results are suspect.

One important change is that the CBGEANT defaults for particle production and tracking are now much lower than the GEANT cuts of 1 MeV for photons, electrons and positrons. Charged particles near 1 MeV have a range of a few millimeters in CsI, allowing them to travel out of a crystal and into the next crystal or nearby insensitive material. Therefore, immediately depositing their energy without tracking them will sharpen the apparent calorimeter resolution. The photon cut is more worrisome because the absorption length of 0.511 MeV annihilation photons in CsI is a few centimeters. The cuts for most regions are listed in the table below, although they are set somewhat higher by calls to GSTPAR in regions which are not near sensitive volumes (e.g., the magnet yoke).

Name of Cut	GEANT (MeV)	CBGEANT (MeV)
`CUTGAM, BCUTE, BCUTM`	1.00	0.15
`CUTELE`	1.00	0.20
`DCUTE, DCUTM`	1.00	0.40
`CUTNEU`	10.00	1.00
`CUTHAD`	10.00	3.00
`CUTMUO`	10.00	2.50

Some of the other interesting changes in defaults include:

- Of the user-defined particle decays in subroutine UGPART, several forbidden channels were replaced by the correct ones. In addition, a number of the particle parameters were updated to the 1990 Particle Data Book values (from the 1986 values).

- Cuts on muons, neutrons and charged hadrons have been lowered.

- Particles reaching $\sqrt{x^2+y^2} > 80$ cm are no longer rejected, allowing effects due to the magnet coil, for example, to be seen.
- Compton, Rayleigh and Moliere scattering, formerly turned off by default, are enabled.
- Muon-nuclear interactions are now included, and nuclear fragmentation is handled by the NUCRIN code.
- Neutrinos are no longer tracked (affecting only CPU time).

A number of experimental effects are now handled in ways expected to be more accurate and/or flexible. These are:

- The JDC track position can now be smeared by a Landau distribution, although this is disabled until tuning has been completed.
- Some of the other JDC signal parameters have been adjusted, including the pulse height and the electrical length of the sense wires. Double and triple pulses have been disabled until they can be adjusted to the data.
- PWC and JDC signal parameters are now accessible through data cards.
- There are new defaults for incoherent noise, and these are different for the FERA and 2282 ADC's. Defaults can be overridden by data card or can be adjusted individually in a data file.
- Coherent noise is now included. The defaults can be overridden, as for the incoherent noise.
- Crystal calibrations now vary randomly. The default is to use a variance of 1%, a value which can be overridden by data card or set for individual crystals in a data file.
- The light produced in the CsI crystals is varied according to dE/dx.

3.3 Platform Support

The code is now compatible with the MIPS compiler used on DECstations. Also, exchange format tape-writing is now supported for UNIX machines, including the DECstations and SPARCstations.

3.4 Event Initialization

The `SETV` and `BWID` data cards, taken together, now allow increased flexibility by allowing the user to set both the width and the centroid of Gaussian vertex distributions, with or without cutting on the physical target volume. For in-flight simulation, a uniform, bounded distribution in Z is possible. Distributions 60–69 are allocated for user definition in an override routine USVERT.

The BIGBANG II generator has been modified, adding intermediate resonances to the extent that they were known and adding a number of known decay channels that were not previously defined. The branching ratios were also updated significantly, which would affect any minimum bias data produced. Also, several of the routines provided for test distributions have been fixed to give particle distributions uniform in solid angle. These are KITWOB, KIUSER and KIXRAY. Finally, the code now contains a particle generator for annihilation in deuterium, KIDEUT.

3.5 Common Blocks

Several common blocks were changed and must be redefined in the user code if used: `CBCNAM`, `CCFLAG`, `CCSETS`, `CCVIEW` and `MCENER`. In addition, several new common blocks were added, and the user should elimininate any conflicts with his/her code: `CBGATE`, `CBJDDF`, `CBJDNS`, `CBJDRS`, `CBJDSG`, `CBKILL`, `CBPWJD`, `CBPWNO`, `CBROT`, `CBTTPR`, `CBVEPR` and `CBXTPR`.

Also, all common blocks in CBGEANT, including those from GEANT and ZEBRA, are explicitly saved. This is because the ANSI-77 standard does not guarantee that common blocks will be saved when leaving a subroutine if the common block is not defined at a higher level in this program branch.

3.6 Error Handling and Debugging

The new version has:

- Optional patches in the subroutine GUSTEP: `DMPSTP` and `DMPVOL` dump information related to the current step and the current volume search tree.
- Modified error-handling routines eliminate some problems encountered under VAX/VMS.
- A simple floating point error handler for the SUN.

3.7 Other

The new version:

- Supports GEANT 3.14 type 'many' volumes through several override routines.
- Has expanded informational banks linked to 'MCIN'.
- Has expanded documentation and scripts in the `$HISTORY` patch.

4 Physics Simulation

4.1 Defined Particles

GEANT defines particle types 1–49 for the so-called stable particles. Their values in GEANT 3.15 are consistent with the 1990 Particle Data Book—with the possible exception of the η lifetime (which is short enough not to affect the time of flight and long enough not to appear in the resolution for the Crystal Barrel).

In addition, the CBGEANT routine UGPART defines a group of mesons needed to describe many of the intermediate states of $\overline{p}$ annihilations. Some mesons are defined with multiple type numbers, with some types restricted to decay only through certain channels. This helps with the rapid accumulation of certain event types; the user may, of course, make further definitions in the MYINIT user routine.

GEANT does not support decays into more than three particles, so one must occasionally make changes to deal with this limitation. In particular, the $f_2(1270)$ is generally listed as having branching ratios into $\pi^+\pi^-\pi^0\pi^0$ and $2\pi^+2\pi^-$. Well within error bars, one could obtain these same states assuming intermediate states of $\rho^+\rho^-$ and $\rho^0\rho^0$, with the former having twice the branching ratio of the latter. Naturally, the phase space distribution of the pions is different between the two cases, but the 1990 Particle Data Book does not rule out the $\rho\rho$ channels, and they would seem to be the more natural channel.

No.	Name	Mass (GeV)	Q/e	Lifetime (sec)	Decay	Percent
1	γ	0.	0	stable		
2	e^+	0.000511	1	stable		
3	e^-	0.000511	-1	stable		
4	ν	0.	0	stable		
5	μ^+	0.105659	1	2.19703E-6	$e^+\nu\nu$	100.0000
6	μ^-	0.105659	-1	2.19703E-6	$e^-\nu\nu$	100.0000
7	π^0	0.134973	0	8.4E-17	$\gamma\gamma$	98.8020
					$e^+e^-\gamma$	1.1980
8	π^+	0.139567	1	2.603E-8	$\mu^+\nu$	100.0000
9	π^-	0.139567	-1	2.603E-8	$\mu^-\nu$	100.0000
10	K_L^0	0.49767	0	5.183E-8	$\pi^+e^-\nu$	19.3500
					$\pi^-e^+\nu$	19.3500
					$\pi^+\mu^-\nu$	13.5500
					$\pi^-\mu^+\nu$	13.5500
					$\pi^0\pi^0\pi^0$	21.5000
					$\pi^+\pi^-\pi^0$	12.3900
11	K^+	0.493646	1	1.2371E-8	$\mu^+\nu$	63.5000
					$\pi^+\pi^0$	21.1600
					$\pi^+\pi^+\pi^-$	5.5900
					$e^+\nu\pi^0$	4.8200
					$\mu^+\nu\pi^0$	3.2000
					$\pi^+\pi^0\pi^0$	1.7300
12	K^-	0.493646	-1	1.2371E-8	$\mu^-\nu$	63.5000
					$\pi^-\pi^0$	21.1600
					$\pi^+\pi^-\pi^-$	5.5900
					$e^-\nu\pi^0$	4.8200
					$\mu^-\nu\pi^0$	3.2000
					$\pi^-\pi^0\pi^0$	1.7300
13	n	0.939566	0	896.		
14	p	0.93827	1	stable		
15	$\overline{p}$	0.93827	-1	stable		
16	K_S^0	0.49767	0	8.922E-11	$\pi^+\pi^-$	68.6100
					$\pi^0\pi^0$	31.3900
17	η	0.5488	0	7.479E-19?	$\gamma\gamma$	39.1000
					$\pi^0\pi^0\pi^0$	31.8000
					$\pi^+\pi^-\pi^0$	23.7000
					$\pi^+\pi^-\gamma$	4.9100
					$e^-e^+\gamma$	0.5000
18	Λ	1.11563	0	2.631E-10	$p\pi^-$	64.2000
					$n\pi^0$	35.8000

No.	Name	Mass (GeV)	Q/e	Lifetime (sec)	Decay	Percent
19	Σ^+	1.18937	1	8.00E-11	$p\pi^0$	51.6400
					$n\pi^+$	48.3600
20	Σ^0	1.19255	0	7.40E-20	$\Lambda\gamma$	100.0000
21	Σ^-	1.19743	-1	1.479E-10	$n\pi^-$	100.0000
22	Ξ^0	1.31490	0	2.90E-10	$\Lambda\pi^0$	100.0000
23	Ξ^-	1.32132	-1	1.639E-10	$\Lambda\pi^-$	100.0000
24	Ω^-	1.67243	-1	8.22E-11	ΛK^-	68.6000
					$\Xi^0\pi^-$	23.4000
					$\Xi^-\pi^0$	8.0000
25	$\overline{n}$	0.939573	0	898.		
26	$\overline{\Lambda}$	1.11563	0	2.631E-10	$\overline{p}\pi^+$	64.2000
					$\overline{n}\pi^0$	35.8000
27	$\overline{\Sigma}^-$	1.18937	-1	8.0E-11	$\overline{p}\pi^0$	51.6400
					$\overline{n}\pi^-$	48.3600
28	$\overline{\Sigma}^0$	1.19255	0	7.4E-20	$\overline{\Lambda}\gamma$	100.0000
29	$\overline{\Sigma}^+$	1.19743	1	1.479E-10	$\overline{n}\pi^+$	100.0000
30	$\overline{\Xi}^0$	1.3149	0	2.9E-10	$\overline{\Lambda}\pi^0$	100.0000
31	$\overline{\Xi}^+$	1.32132	1	1.639E-10	$\overline{\Lambda}\pi^+$	100.0000
32	$\overline{\Omega}^+$	1.67243	1	8.22E-11	$\overline{\Lambda}K^+$	68.6000
					$\overline{\Xi}^0\pi^+$	23.4000
					$\overline{\Xi}^+\pi^0$	8.0000
33	τ^+	1.7841	1	3.04E-13		
34	τ^-	1.7841	1	3.04E-13		
35	D^+	1.8693	1	1.062E-12		
36	D^-	1.8693	-1	1.062E-12		
37	D^0	1.8645	0	4.28E-13		
38	$\overline{D^0}$	1.8645	0	4.28E-13		
39	D_S^+ (F^+)	1.9693	1	4.36E-13		
40	D_S^- (F^-)	1.9693	-1	4.36E-13		
41	Λ_C^+	2.2849	1	1.79E-13		
42	W^+	81.0000	1	9.4E-26		
43	W^-	81.0000	-1	9.4E-26		
44	Z^0	92.4000	0	7.74E-26		
45	Deuteron	1.87561	1	stable		
46	Triton	2.81448	1	stable		
47	He-4	3.72742	2	stable		
48	Geantino	0.	0	stable		
49	He-3	2.81448	2	stable		

Table 1: Standard GEANT particle definitions.

No.	Name	Mass (GeV)	Q/e	Lifetime (sec)	Decay	Percent
50	$b_1^0(1235)$	1.233	0	4.388E-24	$\pi^0\omega$	100.0000
51	$b_1^+(1235)$	1.233	1	4.388E-24	$\pi^+\omega$	100.0000
52	$b_1^-(1235)$	1.233	-1	4.388E-24	$\pi^-\omega$	100.0000
53	$f_2(1270)$	1.274	0	3.560E-24	$\pi^+\pi^-$	56.7000
					$\pi^0\pi^0$	28.3500
					$\rho^+\rho^-$	6.0000
					$\rho^0\rho^0$	3.0000
					K^+K^-	2.4000
					$K_S^0K_S^0$	1.2000
54	$a_2^0(1320)$	1.318	0	5.984E-24	$\pi^+\rho^-$	35.0500
					$\pi^-\rho^+$	35.0500
					$\pi^0\eta$	14.5000
					$\omega\pi^+\pi^-$	10.6000
					K^+K^-	2.4500
					$K_S^0K_S^0$	1.2300
55	$a_2^+(1320)$	1.318	1	5.984E-24	$\pi^+\rho^0$	35.0500
					$\pi^0\rho^+$	35.0500
					$\pi^+\eta$	14.5000
					$\omega\pi^+\pi^0$	10.6000
					$K^+K_S^0$	2.4500
					$K^+K_L^0$	2.4500
56	$a_2^-(1320)$	1.318	-1	5.984E-24	$\pi^-\rho^0$	35.0500
					$\pi^0\rho^-$	35.0500
					$\pi^-\eta$	14.5000
					$\omega\pi^-\pi^0$	10.6000
					$K^-K_S^0$	2.4500
					$K^-K_L^0$	2.4500
57	$\rho^0(770)$	0.7683	0	4.415E-24	$\pi^+\pi^-$	98.7640
					$\pi^+\pi^-\gamma$	1.1100
					$\pi^0\gamma$	0.0790
					$\eta\gamma$	0.0380
					$\mu^+\mu^-$	0.0046
					e^+e^-	0.0044
58	$\rho^+(770)$	0.7683	1	4.415E-24	$\pi^+\pi^0$	99.9550
					$\pi^+\gamma$	0.0450
59	$\rho^-(770)$	0.7683	-1	4.415E-24	$\pi^-\pi^0$	99.9550
					$\pi^-\gamma$	0.0450

No.	Name	Mass (GeV)	Q/e	Lifetime (sec)	Decay	Percent
60	$\omega(783)$	0.78195	0	7.810E-23	$\pi^+\pi^-\pi^0$	88.8000
					$\pi^0\gamma$	8.5000
					$\pi^+\pi^-$	2.2100
					$\pi^0e^+e^-$	0.0590
					$\pi^0\mu^+\mu^-$	0.0096
					e^+e^-	0.0071
61	$\eta'(958)$	0.95757	0	3.160E-21	$\eta\pi^+\pi^-$	44.2000
					$\rho^0\gamma$	30.0000
					$\eta\pi^0\pi^0$	20.5000
					$\omega\gamma$	3.0000
					$\gamma\gamma$	2.1600
					$\pi^0\pi^0\pi^0$	0.1500
62	ETA0	0.5488	0	5.530E-19	$\gamma\gamma$	100.0000
63	ETAP0	0.95757	0	3.160E-21	$\gamma\gamma$	100.0000
64	OMEGA0	0.78195	0	7.810E-23	$\gamma\pi^0$	100.0000
65	OMEGAQ	0.78195	0	7.810E-23	$\pi^+\pi^-\pi^0$	100.0000
66	PI0GG	0.1349734	0	8.400E-17	$\gamma\gamma$	100.0000
67	KSQ	0.497671	0	0.892E-10	$\pi^+\pi^-$	100.0000
68	OMEGA_0	0.78195	0	7.810E-23	PI0GG,γ	100.0000
69	OMEGA_Q	0.78195	0	7.810E-23	PI0GG,$\pi^+\pi^-$	100.0000
70	B0_0	1.233	0	4.388E-24	PI0GG,OMEGA_Q	100.0000
71	ETP9	0.95757	0	3.160E-21	$\eta\pi^0\pi^0$	100.0000
72	$\phi(1020)$	1.019412	0	1.295E-22	K^+K^-	49.5000
					$K^0_LK^0_S$	34.4000
					$\rho^+\pi^-$	4.3000
					$\rho^0\pi^0$	4.3000
					$\rho^-\pi^+$	4.3000
					$\pi^+\pi^-\pi^0$	1.9000

Table 2: CBGEANT particle definitions.

4.2 Defined $p\overline{p}$ Annihilation Channels

Since many of the $\overline{p}$ annihilation channels are not known (this being part of the motivation behind the PS197 experiment), it is clearly not yet possible to entirely simulate minimum bias events. (Even when the number and types of particles in the final states are known, their phase space distributions are affected by the existence of intermediate resonances, and these are usually less well known.) In addition, the branching ratios of some of the intermediate mesons are not well known (e.g., the a_0), and it is not realistic to expect that the CBGEANT code will reflect even the current state of knowledge.

Naturally, annihilation reactions depend on the nature of the target and beam. The channels and branching ratios in BIGTID are for $\overline{p}$ at rest in LH2, and are largely derived from the paper "Antiproton Nucleon Annihilation at Rest: A Data Compilation," P.Blum, H.Koch and G.Buche (September 12, 1989). All previously defined channels (1–29) are still supported; channels 30–59 have been added. Annihilation channels for LD_2 are not yet defined, and the user must specify them for KIDEUT.

Thus, one must take CBGEANT's minumum bias event generation rather lightly. It may be useful as a rough indication of background rates, but one should not expect accurate numbers. The present branching ratios are shown below in Table 3, although one should expect the numbers used in the code to be updated from time to time, as better data becomes available.

However, the user will seldom be generating minimum bias events, but will instead be interested in a specific channel, and (in presumably separate runs) those channels likely to cause background. For these applications, the user will be using his/her own external values of branching ratios.

ID	Fraction: Unnormalized	Channel
1	5.83	$\pi^+\pi^-\pi^0\pi^0$
2	22.34	$\pi^+\pi^-\pi^0\pi^0\pi^0$
3	2.55	$\pi^+\pi^-\pi^0\pi^0\pi^0\pi^0$
4	14.70	$\pi^+\pi^+\pi^-\pi^-\pi^0\pi^0$
5	4.04	$\pi^+\pi^+\pi^-\pi^-\pi^0\pi^0\pi^0$
6	0.02	$\pi^0\pi^0$
7	0.10	K^+K^-
8	1.91	$\pi^+\pi^-\pi^0$
9	1.56	$\rho^0(770)\pi^0$
10	0.24	$f_2(1270)\pi^0$
11	1.46	$\pi^+\pi^+\pi^-\pi^-$
12	1.93	$a_2^+(1320)\pi^-$
13	0.90	$\rho^0(770)f_2(1270)$
14	2.19	$\rho^0(770)\pi^+\pi^-$
15	0.40	$\rho^0(770)\rho^0(770)$
16	0.14	$\omega(783)\pi^+\pi^-$
17	2.48	$\omega(783)\rho^0(770)$
18	1.85	$\omega(783)f_2(1270)$
19	4.93	$\rho^0(770)\pi^+\pi^-\pi^0$
20	0.48	$b_1^+(1235)\pi^-$
21	0.00	$\eta\pi^+\pi^-$
22	0.00	$a_2^+(1320)\pi^-$
23	1.95	$\pi^+\pi^+\pi^+\pi^-\pi^-\pi^-$
24	0.59	$\pi^+\pi^+\pi^+\pi^-\pi^-\pi^-\pi^0$
25	1.24	$\omega(783)\pi^+\pi^+\pi^-\pi^-$
26	0.03	$\eta\pi^+\pi^+\pi^-\pi^-$
27	0.04	$\eta'(958)\pi^+\pi^+\pi^-\pi^-$
28	1.49	$\rho^+(770)\pi^-$
29	3.20	$\rho^+(770)\pi^+\pi^-\pi^-$
30	3.20	$\rho^-(770)\pi^+\pi^+\pi^-$
31	0.00	$\pi^+\pi^+\pi^-\pi^-\pi^0$
32	0.33	$\pi^+\pi^-$
33	0.56	$\pi^0\pi^0\pi^0$
34	1.49	$\rho^-(770)\pi^+$
35	1.93	$a_2^-(1320)\pi^+$
36	1.93	$a_2^0(1320)\pi^0$
37	0.48	$b_1^-(1235)\pi^+$
38	0.72	$\eta\rho^0(770)$
39	0.65	$\eta\pi^0\pi^0$
40	0.31	$\eta'\pi^+\pi^-$

ID	Fraction: Unnormalized	Channel
41	0.02	$\eta\pi^0$
42	0.52	$\omega(783)\pi^0$
43	0.05	$\eta'\pi^0$
44	0.03	$\phi(1020)\pi^0$
45	0.84	$\omega(783)\eta$
46	1.40	$\omega(783)\omega(783)$
47	0.06	$\omega(783)\phi$
48	0.05	$\phi(1020)\pi^+\pi^-$
49	0.06	$K_L^0K_S^0$
50	0.07	$K_S^0K_S^0\pi^0$
51	0.07	$K_L^0K_L^0\pi^0$
52	0.11	$K_S^0K_S^0\omega(783)$
53	0.11	$K_L^0K_L^0\omega(783)$
54	0.23	$K^+K^-\omega(783)$
55	0.23	$K_S^0K_S^0\pi^+\pi^-$
56	0.27	$K_S^0K_L^0\pi^+\pi^-$
57	0.23	$K_L^0K_L^0\pi^+\pi^-$
58	0.26	$K_S^0K^+\pi^-\pi^0$
59	0.26	$K_S^0K^-\pi^+\pi^0$

Table 3: Annihilation branching ratios of $\overline{p}p$ in LH_2, as used in the BIGTID routine of the "Bigbang II" event generator. (Updated in CBGEANT 4.06.)

5 Hardware Simulation

5.1 Volume Definitions

The data card `PRIN 'VOLU'` will generate information about the volumes, along with exact dimensions. This supplements the below table showing volume composition and nesting. (The order shown is *not* the search order.) The daughter volumes of volumes marked with an "$*$" are given in separate tables.

Volumes	Shape	Copies	Media	Type	Description
CB-	BOX		80		Mother volume
CBCO	BOX		81		Concrete floor
MGN-	BOX		15		Box holding magnet
MCO+	TUBE		18		Magnet coil
MCSP	TUBE	5	10		Steel plates in coil
END1	BOX	2	20		Inner, solid door
HOL1	TUBE		15		Hole in inner door
MYO1	BOX	4	10		Yoke: horiz. and vert. slabs
MYO2	BOX	4	10		Yoke: 45° slabs
MYO3	BOX		10		New yoke, top and bottom
MYO4	BOX		10		New yoke, sides
END2	BOX	2	19		Outer, low-density door
HOL2	TUBE		15		Hole in outer door
RAIL	BOX	2	10		Steel support rails
BCM- *	PCON	2	25		BC mother vol./inner alum.
BCCA	TUBE	2	26		BC cables at magnet exit
BCCC	PCON	2	11		Inner cooling collars
JDC- *	PCON		30		JDC mother volume
PWC- *	TUBE		60		PWC mother volume
TGHP	TUBE		51		HP H_2 target mother
THPO	TUBE		53		Al tube
THPE	TUBE	2	53		Al endcaps
LHT- *	PCON		60		LH_2 target mother
GHT- *	TUBE		60		GHT mother, air
VE-	PCON		60		Veto mother volume
VETO	TUBE		90	5	Scintillator
VGUI	TUBE		91		Light guide

BCM- Subvolumes	Shape	Copies	Media	Type	Description
BCDK	TUBE		27		Al annuli (midplane faces)
BCOU	PCON		28		Outer BC support structure
BCA1	PCON		30		Air holding crystals 1–10
BC60		60	30		Azimuthal segment of BCA1
C001-C010	TRAP		22		Crystal cornets 1–10
XT01-XT10	TRAP		21	2	Crystals 1–10
CE01-CE10	TRAP		23		Cornet ends 1–10
CP01-CP10	TRD1		24		Chapeaux 1–10
WS01-WS10	TRD1		90	3	Wavelength shifters 1–10
PD01-PD10	BOX		92	4	Photodiodes 1–10
BCS1	PCON		29		Support structure for 1–10
BCA2	PCON		30		Air volume for xtals 11–13
BC30		30	30		Azimuthal segment of BCA2
C011-C012	TRAP		22		Crystal cornets 11,12
XT11-XT12	TRAP		21	2	Crystals 11,12
C013	PGON		22		Crystal cornet 13
XT13	PGON		21	2	Crystal 13
CE11-CE13	TRAP		23		Cornet ends 1-13
CP11-CP13	TRD1		24		Chapeaux 11-13
WS11-WS13	TRD1		90	3	Wavelength shifters 11-13
PD11-PD13	BOX		92	4	Photodiodes 11-13
BCS2	PCON		29		Support structure for 11-13

JDC- Subvolumes	Shape	Copies	Media	Type	Description
JCEN	TUBE		31		JDC gas (insensitive)
JDG1	TUBE		31		Sensitive gas volume
JDSE		30	31		Azimuthal segment of JDG1
JW01-JW23	TRAP		31	1	Drift cell
JDC1	TUBE		32		Inner cylinder
JDC2	TUBE		33		Outer cylinder
JSD1	TUBE		30		Downstream end material
JDP1	TUBE	1/2	40		Endplate, thin part
JDP2	TUBE	1/2	41		Endplate, thick part
JDCI	TUBE	1/2	37		Inner cooling plate region
JDCO	TUBE	1/2	38		Outer cooling plate region
JAM1	TUBE	1	30		Downstream preamp region
J130		30	30		Azimuthal segment of JAM1
JAMP	BOX	1/2	34		Preamp
JTDN	PCON	1	42		Cables, etc.
JDE1	PCON		35		Support cables, etc.
JDKU	TUBE		39		Swivel support ("kugel")
JSD2	TUBE		30		Upstream end material
JDP1	TUBE	2/2	40		Endplate, thin part
JDP2	TUBE	2/2	41		Endplate, thick part
JDCI	TUBE	2/2	37		Inner cooling plate region
JDCO	TUBE	2/2	38		Outer cooling plate region
JAM2	TUBE	1	30		Upstream preamp region
J230		30	30		Azimuthal segment of JAM2
JAMP	BOX	2/2	34		Preamp
JTUP	PCON	1	42		Cables, etc.
JDE2	PCON		36		Umbilical support cables

PWC- Subvolumes	Shape	Copies	Media	Type	Description
ROHA	TUBE		62		Rohacell
IPWC	TUBE		61		Inner wire layer
ICEL		90	61	1	Azimuthal segment of IPWC
AGAP	TUBE		60		PWC1, PWC2 air gap
OPWC	TUBE		61		Outer wire layer
OCEL		150	61	1	Azimuthal segment of OPWC
ENDL	TUBE		63		G10 end
ENDR	TUBE		63		G10 end

LHT- Subvolumes	Shape	Media	Description
`BLIN`	`PCON`	42	Beamline material
`LHAL`	`TUBE`	53	Aluminum tube
`LHIS`	`TUBE`	55	Mylar insulation
`LHF1`	`TUBE`	56	Iron ring part
`LHFH`	`TUBE`	55/52	Hollow in iron ring (empty or LH_2)
`LHFC`	`TUBE`	57	Copper plate closing ring (1991 or later)
`LHF2`	`TUBE`	56	Iron ring part
`LHMY`	`TUBE`	54	Mylar tube
`LHLH`	`TUBE`	52	Target filling (LH_2 or LD_2)
`LHWI`	`TUBE`	54	Mylar window

GHT- Subvolumes	Shape	Copies	Media	Type	Description
`AFLA`	`PCON`		53		Alum. flaring flange
`BLIN`	`TUBE`		58		Beamline material
`ALCY`	`TUBE`		53		Alum. target cylinder
`HGA1`	`TUBE`		78		H_2 gas
`HGA2`	`TUBE`		78		H_2 gas
`HGA3`	`TUBE`		78		H_2 gas
`ALF2`	`TUBE`		53		Alum. flange
`PPA1`	`TUBE`		53		Alum. PPAC flange
`PPA2`	`TUBE`		70		Plastic PPAC parts
`ALF1`	`TUBE`		53		Alum. PPAC parts
`PPA3`	`TUBE`		70		Plastic PPAC interior
`ALRR`	`TUBE`		53		Fiber retaining ring
`SFI1`	`PCON`		71		Fibers, flaring part
`SCE1`		126	71		Fiber group/segment
`SFI2`	`TUBE`		71		Fibers, cylinder
`SCE2`		126	71		Fiber group/segment

5.2 Materials and Media

Although one can obtain a great deal of information on the material and tracking media definitions with the data card `PRIN 'MATE' 'TMED'`, the table below is included for convenience. Note that these are the CBGEANT materials (*not* tracking media). Materials 1–16 are standard GEANT definitions, while all others are defined in the MATERI subroutine.

Material	Name	Place Used
1	`HYDROGEN`	Target filling
2	`DEUTERIUM`	Target filling
9	`ALUMINUM`	Target, GHT
10	`IRON`	Magnet yokes
11	`COPPER`	BC cooling collars, target part
15	`AIR`	Air inside magnet
18	`AL_COIL`	Magnet
19	`IRON_040`	Magnet
21	`BC_CSI(TL)`	BC
22	`BC_CORNET1`	BC
23	`BC_CORNET2`	BC
24	`BC_CHAPEAU`	BC
25	`BC_ALU1`	BC
26	`BC_CABLES`	BC
27	`BC_ALU2`	BC
28	`BC_ALU3`	BC
29	`BC_BEYOND`	BC
31	`JDC-CO2/ISOBUT`	JDC
32	`JDC-INNER CYL`	JDC
33	`JDC-OUTER CYL`	JDC
34	`JDC-PREAMPS`	JDC
35	`JDC-EXITMASS 1`	JDC
36	`JDC-EXITMASS 2`	JDC
37	`JDC-IN COOLPL`	JDC
38	`JDC-OUT COOLPL`	JDC
39	`JDC-KUGEL`	JDC
40	`JDC-ENDPLATE`	JDC
41	`JDC-ALU_6061`	JDC
42	`JDC-CABLES ETC`	JDC

51	HP H2 GAS	Target
54	MYLAR_LH2	Target
55	MYLAR_INS	Target
56	IRON_100_TARG	Target
58	BEAMLINE	Beamline
61	ARGON/ETHANE	PWC
62	ROH/KAP	PWC
63	G10	PWC
70	PPAC INTERIOR	GHT
78	GHT FIBER LAYER	GHT
78	GHT H2 GAS	GHT
80	CRUDE-CUT AIR	Air outside magnet
81	CONCRETE	Floor
90	SCINTILLATOR	Veto counter
92	SILICON	BC

We now list the tracking media with their names and the number of the material of which they are composed. Material numbers differing from the media number are marked by an "*".

Media	Name	Material
10	Magnet Return Yokes	10
11	BC Cooling Cu	11
15	Default Air	15
18	Aluminium coil	18
19	End2 – 40% Iron	19
20	End1 - 100% Iron	10
21	BC CsI crystal	21
22	BC Cornet1	22
23	BC Cornet2	23
24	BC Chapeau	24
25	BC Alum-Inner	25
26	BC Cables	26
27	BC Alum-Annuli	27
28	BC Alum-Outer	28
29	BC Beyond Chap	29

30	JDC/BC air	15	*
31	JDC CO2:Isobut	31	
32	JDC inner Cyl	32	
33	JDC outer Cyl	33	
34	JDC preamps	34	
35	JDC dwn exit1	35	
36	JDC dwn exit2	36	
37	JDC in cool pl	37	
38	JDC outcool pl	38	
39	JDC kugel	39	
40	JDC thin endpl	40	
41	JDC thickendpl	41	
42	JDC midcool pl	42	
51	High Pressure H2	51	
52	LH2-Target LH2 Fill	1	*
53	Target Aluminum	9	*
54	LH2-Target Mylar	54	
55	LH2-Target Mylar_INS	55	
56	LH2-Target Iron_100	56	
57	LH2-Target Copper	57	
58	LH2-Beamline	58	
59	LH2-Target LD2 Fill	2	*
60	PWC/LH2 mother (air)	15	*
61	PWC Ar/Ethane	61	
62	PWC Rohacell	62	
63	PWC G10 endplate	63	
70	GHT - PPAC INTERIOR	80	
71	GHT - FIBER LAYER	80	
78	GHT - 1 ATM. H2 prec	80	
80	Low-accuracy Air	80	
81	Crude Concrete	81	
90	Scintillator	90	
91	Light guide	90	*
92	Silicon diode	92	

5.3 Detector Subcomponents

5.3.1 Liquid Hydrogen Target

The liquid hydrogen target is implemented fairly accurately, excepting details like the bulging of the Mylar windows, the smearing of the superinsulation into a continuous medium and the omission of the Vespel tapering from the iron ring to the Mylar cylinder. Because the Mylar hydrogen vessel is so thin, the tracking resolution would normally have to be set to a rather small value. Therefore, its thickness was increased in the simulation, and its density reduced accordingly. Of course, the Sternheimer density effect changes the energy loss calculation somewhat, but the Mylar here is so thin that this should not matter.

There was a design change made in the target before the 1991 runs: a 2-mm thick copper disk was added to close the "iron" ring, allowing it to be used for cooling. This is supported in the Monte Carlo through the `TGLH` data card, allowing the user to specify the year. The same card allows the user to specify the target filling as being either LH_2 or LD_2, although this affects only the interactions of the annihilation products—not the annihilation itself.

5.3.2 PWC

There have been three pairs of PWC's to date; only the second design is supported in CBGEANT. The first design had more wires in each chamber than does the current design and they are therefore quite distinct. The second and third designs are more similar.

The PWC geometry is implemented straightforwardly, except that the wires are not included in the gas, and the Rohacell, Mylar and aluminum of the walls have been homogeneously smeared together. The signal cables have been included only approximately.

5.3.3 GHT

The GHT and PPAC geometries have been implemented, although some dimensions may be slightly inaccurate and a number of minor features (e.g., the Mylar window) have been ignored or simplified. The scintillating fibers have been smeared into a continuous media of the correct average density. This is done for two regions: the cylinder around the target and the part which flares out to larger radius before the fibers are spliced to non-scintillating fibers. These regions have then been divided into 126 sectors.

5.3.4 JDC

The JDC is geometrically complex, with its mass sufficient to cause significant γ-ray conversion and charged particle energy loss. The mass and composition of most regions are known and are usually implemented accurately, to the extent possible with smearing. An exception would be the chamber sense and high voltage wires, which have not been included at all, although their scattering is *not* negligible compared to the gas. Including them properly would require some effort, since it would not be correct to simply smear them with the gas.

One type of region which has not been entirely smeared corresponds to the 30 preamps at either end of the chamber. Viewed edgewise, these are quite thick; unfortunately, the products from a converting, high-energy γ ray will usually traverse the full width and lose a non-negligible amount of energy. Azimuthally smearing the preamps into a ring would give a lower energy loss. Thus, each preamp is represented by a distinct region, although the components of the preamp are still smeared over a rectangular solid volume.

5.3.5 Barrel Calorimeter

The geometry of the barrel has been implemented rather accurately. A small exception would be the type #11 crystals, at which there is a transition from 30 crystals per ring to 60 crystals per ring. Since these crystals are not trapezoidal in cross-section, GEANT does not support their shape, and they have to be approximated. At one edge, the gap to the next crystal is too large in some places, while it slightly overlaps the next crystal in others. The other major calorimeter approximations are radially beyond the crystals, such as smearing of the crystal chapeaux and parts of the support structure, which probably have relatively little effect.

5.3.6 Magnet

Physically, the magnet simulation should be reasonably accurate. The aluminum, epoxy and water in the coil are smeared together, but the steel plates between the coil pancakes and the calorimeter support rails are included separately. The components of the yoke are all included, except that part of the doors, where steel plates alternate with air, is smeared into a region of reduced density. There is even some concrete under the magnet for the benefit of any neutrons which may feel inclined to scatter (although shielding blocks and other material external to the magnet have been omitted).

5.3.7 Veto Counter

The veto counter can be included physically, although it is the user's responsibility to position it. As of CBGEANT 4.06/03, however, there is still no provision to digitize signals in it.

5.4 Magnetic Field

There are some uncertainties in the homogeneity of the magnetic field; the various expressions available so far do not obey Maxwell's equations closely. If one chooses the CBGEANT option for Runge-Kutta tracking in the central part of the detector, the equations used for the field are:

$$B_x = B_0 x\,(2c_1 z + c_2)$$

$$B_y = B_0 y\,(2c_1 z + c_2)$$

$$B_z = B_0\left[1 + c_1\left(r^2 - 2z^2\right) - 2f_2 z\right], \text{ where}$$

$$c_1 = 1.016 \times 10^{-5} \text{ and } c_2 = 0.293 \times 10^{-5}.$$

B_0 is the field at the center of the detector (normally 1.5 Tesla).

In any case, the magnetic field in the coil and beyond is not accurate; it is taken as being constant and equal to the central value in each region, which is clearly not true in the coil or magnet doors. A small exception is the outer yoke, where the sign of the field is taken to be opposite that of the central field. However, one does not expect much effect from details in the field beyond the calorimeter, since most particles will not be deflected back into the detector.

5.5 Detector Digitization

5.5.1 PWC

The PWC digitization algorithm PWDIGI is fairly simple. It applies a constant efficiency (96% for both chambers), together with a constant probability of triggering wires adjacent to a hit and a constant probability of there being noise hits. One would expect that, in reality, chamber efficiency would be greater for heavily ionizing tracks, and for tracks with large azimuthal and/or axial directional components. Such tracks would presumably also have an increased probability of triggering adjacent wires. The parameters affecting the PWC signal are accessible by data card.

5.5.2 GHT

Data from the GHT is not currently digitized. In principle, one could take the recorded entrance and exit locations for a particle, determine the distance which a straight line path would take through the five active fibers in a sector and scale the energy loss in the continuous media accordingly. This would be a simple algorithm to write, and reasonably fast to execute.

5.5.3 JDC

There are two separate algorithms for producing signals in the JDC with CBGEANT. The physical Monte Carlo is somewhat slow, but simulates the detailed physical processes that generate a signal. The `FASTJDC` algorithm is much simpler. It takes the entrance and exit points on a JDC cell, finds the shortest drift time along the straightline connection of these points and then applies various distortions. Most testing has been done with the `FASTJDC` option so far. The parameters affecting the `FASTJDC` calculation of the JDC signal are accessible by data card.

5.5.4 Barrel Calorimeter

There are a number of small corrections made to the energy deposit in the CsI in calculating the appropriate signal. There is compensation for variations in dE/dx which affect the light output, but this is not done quite right: GEANT uses a restricted energy loss, while the correction function in the literature is for unrestricted energy loss. Fortunately, this has only a minor effect. The correction that exists for the time of energy deposit relative to the ADC gate is also small, presumably affecting mostly muon decays (neutrons do not decay in GEANT). Finally, energy deposits in the wavelength shifter and photodiode are scaled and added to the energy deposit in the CsI (although the scaling factor for the wavelength shifter is not yet accurately known).

The main uncertainty in the light output corrections is that the light collection efficiency varies considerably (several per cent) along the axis of a crystal, and the variation varies from one crystal to the next. The measurements of the crystal compensation were made with a photomultiplier tube, rather than a silicon photodiode, and are not entirely applicable. It is likely, however, that one cannot fully simulate the calorimeter without handling this problem in some way, since the variation is of the same order as the resolution. A constant smearing is not appropriate because is fails to account for the larger variation which higher energy electromagnetic showers and heavy charged particles will tend to sample (crystal compensation being worse near the chapeaux).

There are a number of data card options to impose crystal calibration variations, incoherent noise and coherent noise. (By default, there is a 1% variation in the crystal calibrations.) These are straightforward and the data cards are discussed elsewhere.

5.6 Particle Production and Tracking Thresholds

In a condensed history Monte Carlo such as GEANT, it is critical that the energy cuts for discrete particle production and tracking be chosen carefully: using the code as a black box is a prescription for disaster. To use thresholds that are too high results in the loss of important information, such as the leakage of energy from one crystal into adjacent crystals in a calorimeter, or the loss of tracking resolution in a gaseous tracking detector due to the low energy e^-'s and γ's that are produced along a track and smear it. On the other hand, low energy electrons are expensive to track, both because their number nearly diverges at low energy and because their scattering and deflection in a magnetic field is large, so that step lengths in a Monte Carlo must be small. This is especially true for GEANT where the angular step in the magnetic field is a quantity on which the user sets an upper limit.

Fortunately, CPU time is not as sensitive to these cuts as one might expect, since, although copious, low energy particles usually do not have to be tracked very far (excepting neutrons and neutrinos). Decreasing most energy cuts by a factor of ten from the GEANT defaults costs a factor of two in CPU time. Thus, there is not too much point in worrying over extremely fine adjustments of these parameters: one should set them at a reasonably safe level.

While it is not possible to choose the appropriate energy thresholds with confidence prior to runs verifying that the measured physics is stable with respect to moderate variations in these thresholds, a rough rule of thumb would be that the practical range or attenuation length of the secondary particles at the threshold should not much exceed the resolution or smallest dimensions of the sensitive portions of the device. One must examine some specific cases to make sense of this vague language.

1. Consider CsI crystals with dimensions of order 30 cm x 4.5 cm x 3 cm, separated by 0.1 g/cm^2 thick titanium foil. If one sets the thresholds at energies below which the particles do not often reach an adjacent crystal, this is fairly safe. Of course, some low energy δ-rays from energy deposition near a crystal face might leak energy into the titanium without this being simulated if the energy loss is treated as continuous, but the fraction of secondaries affected would be small.

2. In the JDC drift chamber's 1.5 Tesla magnetic field, charged secondaries tend to spiral along the field lines, so one would compare the diameter of the spiral and the range to the r-ϕ resolution, and also compare the range with the Z resolution. Since the spiral's diameter is $D < 0.44$ P cm/(MeV/c), an r-ϕ resolution of 100 μm or 0.00002 g/cm^2 of CO_2 would correspond to a momentum of $P = 0.023$ MeV/c or a kinetic energy of $T = 0.0005$ MeV for $e^{+/-}$. A range of 100 μm, though, corresponds to $T = 0.0025$ MeV for $e^{+/-}$. A Z resolution of 0.75 cm would correspond to 0.75 cm $\times$ 0.002 g/cm^3 = 0.0015 g/cm^2, implying $P = 0.155$ MeV/c and that $T = 0.024$ MeV for an $e^{+/-}$. An attenuation length of 100 μm would correspond to 0.003 MeV X rays. Cuts this low, however, are probably not practical with GEANT. Thus, one does not expect to be able to use GEANT by itself for precise drift chamber modeling.

Neutron cuts are rather difficult to guess, since neutrons have a long mean free path, even at low energies. Since they will often deposit their energy far away from the original PED, an energy cut should be lower than that used to define a separate PED, and smaller than the typical energy resolution for a crystal. However, because they are less copious than $e^+/e^-/\gamma$'s, and because they tend to arise mostly from pion impacts for which there is widely varying energy leakage from a crystal, it is probably not necessary to set the threshold extremely low.

6 Running the Code

6.1 Input

6.1.1 Data Cards

GEANT itself defines many data cards, and a number of others are defined in the UGFFGO subroutine of CBGEANT. An attempt will first be made to read data cards from logical unit 99, except on CERNVM. If this unit is not assigned to an existing file, the cards will be read from standard input; the actual input is done by FFREAD.

The card list below includes all of the CBGEANT-defined cards, as well as a few of the more commonly used GEANT cards (marked with an "*"); see the GEANT manual for a complete list. The list gives the keyword, argument types, the common block and variables affected by each argument, and the default values and meanings for each variable. The convention for the argument types is that i1,i2,i3... are integer; r1,r2,r3... are real; L1,L2,L3... are logical; c1,c2,c3... are four-character Hollerith variables and s(n) is a character string of length n.

Name	Type	Common	Variable	Default	Description
AGHT	r1	CBTGGH	ANGGHT	0.	+Y to GHT sector 1 center (deg)
AJDC	r1	CBPWJD	ANGJDC	0.97	+Y to JDC sector 1 center (deg)
ALOR	r1	CBPWJD	ANGLOR	-7.0	JDC Lorentz angle (deg)
APWC	r1	CBPWJD	ANGPWC(1)	0.	+Y to PWC1 wire 0 (deg)
	r2		ANGPWC(2)	0.	+Y to PWC2 wire 0 (deg)
BCCC	L1	CBBCPR	LBCCC	ON	ON: include BC cooling collars
BWID	r1	CBTTPR	WIDIN(1)	0.	Vertex σ_X, SETV=90–93
	r2		WIDIN(2)	0.	Vertex σ_Y, SETV=90–93
	r3		WIDIN(3)	0.	Vertex σ_Z, SETV=90–91
					$(Z_{max} - Z_{min})/2$, SETV=92–93
CCUT	L1	CBMCUT	LCECUT	OFF	ON: Particle thresholds constant
DEBU*	i1	GCFLAG	IDEMIN	0	First event to debug
	i2		IDEMAX	0	Last event to debug
	i3		ITEST	0	Debug each ITESTth event
DIGI	c1–c5	CCFLAG			Subdetector digitized iff .T.:
			'GH '->QDIGI(1)	.F.	GHT
			'PW '->QDIGI(2)	.T.	PWC
			'JD '->QDIGI(3)	.T.	JDC
			'BC '->QDIGI(4)	.T.	BC
			'VE '->QDIGI(9)	.F.	Veto counter

GEOM	c1–c9	CCFLAG			Use subdetector iff .T.:
			'GH '->QGEOM(1)	.F.	GHT
			'PW '->QGEOM(2)	.T.	PWC
			'JD '->QGEOM(3)	.T.	JDC
			'BC '->QGEOM(4)	.T.	BC
			'MG '->QGEOM(5)	.T.	Magnet
			'TGLH'->QGEOM(7)	.T.	LH$_2$
			'TGHP'->QGEOM(8)	.F.	HP
			'VE '->QGEOM(9)	.T.	Veto counter
GET	c1...	CCFLAG			
			'VERT'->QPRIN(KVERT)	.F.	CALL GPVERT iff .T.
			'VERT'->QZGET(KVERT)	.F.	
			'KINE'->QZGET(KKINE)	.F.	Get KINE bank iff .T.
			'HITS'->QZGET(KHITS)	.F.	Hits read from tape iff .T.
			'DIGI'->QZGET(KDIGI)	.F.	Digits read from tape iff .T.
			'SETS'->QZGET(KSETS)	.F.	
			'VOLU'->QZGET(KVOLU)	.F.	
			'GEOM'->QZGET(KGEOM)	.F.	
			'MATE'->QZGET(KMATE)	.F.	
			'TMED'->QZGET(KTMED)	.F.	
			'CXYZ'->QZGET(KCXYZ)	.F.	
			'JXYZ'->QZGET(KJXYZ)	.F.	
			'ROTM'->QZGET(KROTM)	.F.	
			'PART'->QZGET(KPART)	.F.	
HDTG	i1		IVHDTG	0	Trigger #; not used
IDEN	L1	CCFLAG	LIDENT	OFF	Routines print ID msg. iff .T.
IFLD	i1	CCSETS	IFLDPJ	3	IFIELD for target, PWC and JDC
JDC	s(20)	CCJDCD	FILEJ	' '	JDC data file name
JDNS	r1		RSIGJD	0.1	JDC signal fractional variation
	r2		RNOIJD	50.0	JDC signal noise
JDRS	r1	CBJDRS	SIGXJD	0.000	JDC Landau smearing in X (cm)
	r2		SIGYJD	0.000	JDC Landau smearing in Y (cm)
	r3		ZRESJD	0.70	JDC Gaussian smearing in Z (cm)
	r4		ZMAXJD	1.40	JDC smearing limit in Z (cm)
JDSG	i1	CBJDSG	KTHRES	100	JDC threshold
	r1		AVALA	0.15	JDC signal height normalization
	r2		AVALS	0.2	JDC avalanche distribution width
	r3		PDOUBL	0.0	JDC double signal fraction
	r4		PTRIPL	0.0	JDC triple signal fraction
	r5		HWLEN	20.	JDC sense wire half length (cm)
KILL	i1	CBKILL	NPKILL	1	# particle types to kill
	i2		IDKILL(1)	4	First particle ID to kill
	i3-i11		IDKILL(2-10)	0	Other particle ID's to kill

KINE	i1	GCKINE	IKINE	3	Particle generators
KINE 2					KIUSER one-particle generator
	r1		PKINE(1)		Particle ID
	r2		PKINE(2)		Lower $\|P\|$ limit (GeV/c)
	r3		PKINE(3)		Upper $\|P\|$ limit (GeV/c)
	r4		PKINE(4)		θ (degrees); isotropic if < 0
	r5		PKINE(5)		ϕ (degrees); isotropic if < 0
KINE 3					KIBIBA generator
	r1		PKINE(1)	1.	1: Hit/miss Monte Carlo 0: Normal Fowl Monte Carlo
	r2		PKINE(2)	0.	P_z of $\overline{p}$ beam (GeV/c)
	r3		PKINE(3)	-8.	$< 0.$: BIGTID decay channel (1–30) $= 0.$: Event from weighted table $> 0.$: # of listed particles
	r4		PKINE(4)		ID of first particle
	r5–r7		PKINE(5-7)		$\|P\|$, θ, ϕ for r3=1; isotropic if $\theta < 0$ or $\phi < 0$
	r5–r9		PKINE(5-9)		Particle types for r3> 1.
KINE 31					KIDEUT (LD_2) generator
	r1		PKINE(1)	1.	1: Hit/miss Monte Carlo 0: Normal Fowl Monte Carlo
	r2		PKINE(2)	0.	P_z of $\overline{p}$ beam (GeV/c)
	r3		PKINE(3)	-8.	$> 1.$: # of listed particles
	r4		PKINE(4)		ID of spectator particle
	r5		PKINE(5)		ID of first produced particle
	r6–r10		PKINE(6-10)		Particle types
KINE 50					KIRFQ isotropic γ generator
			PKINE(1)		Energy of γ's (0-100 GeV)
			PKINE(2)		Number of γ's (1-1000)
KINE 99					KITWOB 2-body generator for $\overline{p}p$; uniform for $12° < \theta < 168°$
	r3		PKINE(3)		First particle ID
	r4		PKINE(4)		Second particle ID
MAGN	r1	CCSETS	FLDM	-15.	B field at origin (KG)
PREC	c1–c9	CCFLAG			Precise tracking iff .T.:
			`'GH  '->QPREC(1)`	.F.	GHT
			`'PW  '->QPREC(2)`	.F.	PWC
			`'JD  '->QPREC(3)`	.F.	JDC
			`'BC  '->QPREC(4)`	.F.	Barrel calorimeter
			`'MG  '->QPREC(5)`	.F.	Magnet
			`'TGLH'->QPREC(7)`	.F.	LH_2 or LD_2 target
			`'TGHP'->QPREC(8)`	.F.	High Pressure target
			`'VE  '->QPREC(9)`	.F.	Veto counter

PRIN	c1–c13	CCFLAG			Extra printing iff .T.
			'VERT'->QPRIN(KVERT)	.F.	CALL GPVERT
			'KINE'->QPRIN(KKINE)	.F.	CALL GPKINE
			'HITS'->QPRIN(KHITS)	.F.	CALL GPHITS
			'DIGI'->QPRIN(KDIGI)	.F.	CALL GPDIGI
			'SETS'->QPRIN(KSETS)	.F.	CALL GPSETS
			'VOLU'->QPRIN(KVOLU)	.F.	CALL GPVOLU
			'GEOM'->QPRIN(KGEOM)	.F.	
			'MATE'->QPRIN(KMATE)	.F.	CALL GPMATE
			'TMED'->QPRIN(KTMED)	.F.	CALL GPTMED
			'CXYZ'->QPRIN(KCXYZ)	.F.	CALL GPCXYZ (each step)
			'JXYZ'->QPRIN(KJXYZ)	.F.	ISWIT(10)=1: save JXYZ data
			'ROTM'->QPRIN(KROTM)	.F.	CALL GPROTM
			'PART'->QPRIN(KPART)	.F.	CALL GPPART
PWCD	i1	CBPWJD	NDPWC(1)	2	PWC1 design type
					0: Exclude PWC1
					1: 1989 PWC1 (not done)
					2: 1990–1991 PWC1
	i2		NDPWC(2)	2	PWC2 design type
					0: Exclude PWC2
					1: 1989 PWC2 (not done)
					2: 1990–1991 PWC2
PW1S	i1	CBPWNO	NWPW1	1	Max. PWC1 hits from stat. noise
	r1		PWNOI1	0.10	Prob. possible noise hit occurs
	r2		PWCHE1	0.96	PWC1 chamber efficiency
	r3		PWNXT1	0.20	Prob. hit gives adjacent hit
PW2S	i1	CBPWNO	NWPW2	1	Max. PWC2 hits from stat. noise
	r1		PWNOI2	0.10	Prob. possible noise hit occurs
	r2		PWCHE2	0.96	PWC2 chamber efficiency
	r3		PWNXT2	0.20	Prob. hit gives adjacent hit

RAWD	i1	CCRAWD	IORAWD	20	Unit for raw data output
	c1-c20				Banks to dump, etc.
			'ALL '->QRAWD(KALL)	.F.	Output EVHD, RPWC, RJDC, RBCL, RBCF, RMCB if .T.
			'EVHD'->QRAWD(KEVHD)	.F.	Output EVHD if .T.
			'RRAW'->QRAWD(KRRAW)	.F.	Output RRAW if .T.
			'RBCF'->QRAWD(KRBCF)	.F.	Output RBCF if .T.
			'RBCL'->QRAWD(KRBCL)	.F.	Output RBCL if .T.
			'RGHC'->QRAWD(KRGHC)	.F.	Output RGHC if .T.
			'RGHF'->QRAWD(KRGHF)	.F.	Output RGHF if .T.
			'RJDC'->QRAWD(KRJDC)	.F.	Output RJDC if .T.
			'RJDF'->QRAWD(KRJDF)	.F.	Output RJDF if .T.
			'RPWC'->QRAWD(KRPWC)	.F.	Output RPWC if .T.
			'RMON'->QRAWD(KRMON)	.F.	Output RMON if .T.
			'ROTR'->QRAWD(KROTR)	.F.	Output ROTR if .T.
			'RMCB'->QRAWD(KRMCB)	.F.	Output RMCB if .T.
			'DISK'->QRAWD(KDISK)	.F.	Write to disk if .T. (Only if data is *never* to be copied to tape.)
			'TAPE'->QRAWD(KTAPE)	.T.	Write to tape if .T. (Used if data might be copied to tape.)
			'NATI'->QRAWD(KNATI)	.T.	NATIve Zebra format if .T.
			'EXCH'->QRAWD(KEXCH)	.F.	EXCHange Zebra format if .T.
			'ALFA'->QRAWD(KALFA)	.F.	ALFA Zebra format iff .T.
RMAX	r1	CCTRAK	RMAX	500.	Kill particles if $\sqrt{x^2+y^2}$ >RMAX
RNDM *	i1	GCFLAG	NRNDM(1)		Random seed #1
	i2		NRNDM(2)		Random seed #2
SETS	c1–c5		Make regions sensitive iff .T.		
			'GH '->QSETS(1)	.F.	GHT iff .T.
			'PW '->QSETS(2)	.F.	PWC iff .T.
			'JD '->QSETS(3)	.F.	JDC iff .T.
			'BC '->QSETS(4)	.F.	BC iff .T.
			'VE '->QSETS(5)	.F.	VETO iff .T.

SETV	i1	CCVERT	IVERTI	0	Sets vertex type:
					0: Vertex = (0,0,0)
					2: Asterix
					3: Asterix, target cut
					4: CB TGH2 (obsolete)
					5: CB TGH2, target cut (obsolete)
					6: CB TGHP
					7: CB TGHP, target cut
					8: CB TGLH
					9: CB TGLH, target cut
					60–69: User defined in USVERT
					80: r1–r3 are σ's around (0,0,0)
					81: As for 80, with target cut
					90: σ's from BWID card
					91: As for 90, with target cut
					92: σ_x, σ_y, ΔZ from BWID card
					Dist. is uniform in Z over $\pm\Delta Z$
					93: As for 90, with target cut
	r1		VERTIN(1)		$\overline{x}$ for uncut vertex dist.
	r2		VERTIN(2)		$\overline{y}$ for uncut vertex dist.
	r3		VERTIN(3)		$\overline{z}$ for uncut vertex dist.
STAT *					More GEANT statistics banks
SWIT	i1	GCFLAG	ISWIT(1)	0	1: Print primary vertex/kinematics
	i2		ISWIT(2)	0	
	i3		ISWIT(3)	0	
	i4		ISWIT(4)	0	
	i5		ISWIT(5)	0	
	i6		ISWIT(6)	0	PRIN JXYZ prints i6th track;
					(use i6=0 to get all tracks)
	i7		ISWIT(7)	0	PRIN JXYZ prints i7th track
	i8		ISWIT(8)	0	PRIN JXYZ prints i8th track
	i9		ISWIT(9)	0	
	i10		ISWIT(10)	0	1: store the JXYZ structure
					10: drop the JXYZ structure
TGLH	c1	CBTTPC	TGFILL	'LH2 '	'LH2 ': TGLH target fill is LH$_2$
					'LD2 ': TGLH target fill is LD$_2$
	c2		TGYEAR	'1990'	'1989' or '1990': No cooling circuit
					'1991' or '1992': Cooling circuit
TIME *	r1	GCTIME	TIMINT	System	Time left after initialization
	r2		TIMEND	1.	Time required for termination
	i1		ITIME	1	Test every ITIME events
					Uses KERNLIB routine TIMEL:
					non-existent on SUN and others

VCTR	r1	CBVEPR	PVETO(1)	200.0	Z at π counter face (cm)
	r2		PVETO(2)	1.E6	Veto threshold (MeV); not used
XCAF	s(20)	CBXTPR	XCFILE	' '	Crystal calibration file name
XCVA	r1	CBXTPR	CONXVA	0.01	σ of crystal calibration
XNFE	r1	CBXTPR	CONXF(1)	0.060	Fera RMS coherent noise (MeV)
			CONXF(2)	0.500	Fera RMS incoherent noise (MeV)
XNFF	s(20)	CBXTPR	XFFILE	' '	Fera noise file name
XN22	r1	CBXTPR	CONX2(1)	0.060	2282 RMS coherent noise (MeV)
	r2		CONX2(2)	0.250	2282 RMS incoherent noise (MeV)
XN2F	s(20)	CBXTPR	X2FILE	' '	2282 noise file name
XRAY	i1	CCVERT	IVXRAY	0	Number of protonium X rays
ZTAR	r1	CBTTPR	ZTARG	0.	Z at TGLH target center (cm)

Notes:

1. Use correct letter case with UNIX file names.
2. If the `DIGI`, `GEOM` or `SETS` cards are used, the defaults for that card are not used. *All* desired detector elements must then be explicitly listed.
3. Use of the `GEOM`, `PWCD`, `SETS`, `DIGI` and `RAWD` cards can affect the values set by the others. UGCHCK attempts to set all of these switches to reasonable, self-consistent values. One should check the program's informational output to unit `LOUT` to verify that they are set to what is actually desired.
4. For the `GEOM` card, the `GH` and `PW` options are mutually exclusive, as are the `TGLH` and `TGHP` options.
5. For the Hollerith variables, use upper case letters and include trailing blanks (some machines may not require this).

6.1.2 Drift Chamber Files

The drift time file associated with the `JDC` data card is an ASCII file. To obtain meaningful JDC tracking results, this file must have been produced for the appropriate magnetic field, gas mixture, pressure and JDC geometry, including electrostatics. The files included in the `DFILES` patch of 4.06 supercede previous tables, being produced with the correct magnetic field direction, a gas density appropriate to CERN, the final JDC geometry and the voltages used from December 1990 through, at least, December 1991. They differ from the files distributed with CBGEANT 4.04 and 4.05 in that they contain additional header information.

The Lorentz angle specified with the `ALOR` data card should correspond as closely as possible to the drift line angles corresponding to the drift time file. These vary with magnetic field, among other things. An incorrect Lorentz angle will cause inaccuracies in the drift cell boundaries with the `FASTJDC` patch, causing errors in the calculation of the minimum drift time from a track to the sense wire when that minimum occurs near the boundary—usually when the azimuthal component of a track dominates.

A modified version of the GARFIELD program was used to calculate the drift time files—the modifications being needed to provide the diffusion data and to put the file into the expected format. Header information has been added "by hand." Drift values are given on an r, ϕ grid which is 400×25. The accuracy is uncertain in the inner and outer-most cells because of the somewhat unknown and variable electrostatic charging on nearby insulators.

6.1.3 Barrel Calorimeter Files

The files associated with the `XCAF`, `XNFF` and `XN2F` data cards are formatted and are input with a list-directed read. Entries are ordered by a compound crystal number defined as: $N = I_\phi + (I_\theta - 1) * 60$, with $I_\theta = 1, 2, 3, \ldots, 26$, where the $I_\theta = 1$ crystal ring is centered 15° from the $+Z$ axis. $I_\phi = 1, 2, 3, \ldots, 60$ for $I_\theta \epsilon [4, 23]$ with the $I_\phi = 1$ crystal starting at $\phi = 0°$; $I_\phi = 2, 4, 6, \ldots, 60$ for $I_\theta = 1, 2, 3, 24, 25, 26$ with the $I_\phi = 2$ crystal starting at $\phi = 6°$. There are thus $60 \times 26 = 1560$ entries in these files, of which $30 \times 6 = 180$ are unused. The files begin with a three line header, with up to 79 characters per line for file identification and comments. These are echoed to the `LOUT` unit by BCSET during CBGEANT initialization. (A blank is added in front of each comment line on the output.)

6.2 User Routines

There are three basic CBGEANT user routines: MYINIT, MYEVNT and MYLAST. These are often used for histogram definition, event-by-event filling and output. Dummy versions are included with CBGEANT, and it is up to the user to override them with active subroutines. All three are called through CBGEANT.

MYINIT This user initialization subroutine is called by UGINIT once after the other initialization has been done, excepting calls to UGINFO (which writes out miscellaneous, pertinent information) and UGZRUN (which fills the lookup tables). Typically, one would use MYINIT to book histograms, initialize user variables and/or modify particle decays. (Note that only decays to two or three particles are allowed by GEANT.)

MYEVNT This subroutine is called once after each event. The call is at the beginning of GUOUT. Typically, one would fill histograms and/or extract other information about an event here.

MYLAST This subroutine is for whatever termination functions are needed by the user. It is called once, by UGLAST, after most other clean-up work is done, including the termination of GEANT. Only the raw data output file and the ZEBRA buffers have yet to be closed. Typically, one would save histograms and/or record other user-calculated information here.

There are also two routines intended to be overridden by the user as needed for event initialization: KIUSER and USVERT. These are for setting event kinematics and vertex position, respectively.

We now give a simple, but practical, example of user routines. One thousand 50-MeV γ's were started isotropically from a beam spot of finite size at the spectrometer center; the calibration uncertainty of the crystals was set to zero. The data cards were:

```
LIST
TRIG 1000
TGLH 'LH2 ' '1991'
PWCD 2 2
JDC 'jdc15.dat'
XCVA 0.00
SETV 91 0. 0. 0.
BWID 0.3 0.3 0.6
KINE 2 1. 0.050 0.050 -1. -1.
RAWD 20 'TAPE' 'EXCH' 'EVHD' 'RPWC' 'RJDC' 'RBCL' 'RBCF' 'RMCB'
END
```

The dummy user routines in the `cbuser` patch of CBGEANT include the code in this example as a comment section.

```
      SUBROUTINE MYINIT
*
+SEQ,CCVERN.
+SEQ,GCFLAG.
*
      PARAMETER LUN17=17
      CHARACTER*11 NAME
*
* Set up output files:
      IENER = IFIX(1000.*PKINE(2))
      IF (IENER.LT.10) THEN
         WRITE (NAME,801) IENER
      ELSEIF (IENER.LT.100) THEN
         WRITE (NAME,802) IENER
      ELSEIF (IENER.LT.1000) THEN
         WRITE (NAME,803) IENER
      ELSEIF (IENER.LT.10000) THEN
         WRITE (NAME,804) IENER
      ELSE
         STOP ’ ERROR: UNEXPECTEDLY HIGH GAMMA ENERGY’
      ENDIF
*
      OPEN (UNIT=LUN17, FILE=NAME, FORM=’FORMATTED’, STATUS=’UNKNOWN’)
      REWIND (LUN17)
      CALL HOUTPU(LUN17)
      CALL HERMES(LUN17)
*
* Book histograms:
      DNLIM = 0.10*1000.*PKINE(3)
      UPLIM = 1.10*1000.*PKINE(3)
      CALL HBOOK1(1,’SUM BC HIT ENERGY, ALL$’, 100,DNLIM,UPLIM,0.)
      CALL HBOOK1(2,’SUM BC HIT ENERGY, HITS > 5 MEV$’,
     &            100,DNLIM,UPLIM,0.)
      CALL HBOOK2(21,’TOTAL BC ENERGY VS INITIAL THETA$’,
     &            60,0.,180.,50,DNLIM,UPLIM,0.)
      RETURN
 801  FORMAT (’gam000’,I1,’.out’)
 802  FORMAT (’gam00’, I2,’.out’)
 803  FORMAT (’gam0’,  I3,’.out’)
 804  FORMAT (’gam’,   I4,’.out’)
      END
```

```
      SUBROUTINE MYEVNT
+SEQ,GCFLAG.
+SEQ,GCKINE.
+SEQ,MCENER.
      REAL EXTAL(1560)
*
* Clear xtal energy deposits:
      DO 5 I=1,1560
         EXTAL(I) = 0.
 5    CONTINUE
*
* Loop over all tracks and hits:
      ITRACK = 0
 10   CONTINUE
      ITRACK = ITRACK + 1
      IF (ITRACK.GT.50) GOTO 950
      NHITS = ITRKMC(ITRACK,0)
      IF (NHITS.LT.0 .OR. NHITS.GT.100) GOTO 950
*
      DO 15 IHIT=1,NHITS
         EHIT = ENTKMC(ITRACK,IHIT)
         NCOMP = ITRKMC(ITRACK,IHIT)
         IF (NCOMP.LT.1 .OR. NCOMP.GT.1560) GOTO 950
         EXTAL(NCOMP) = EXTAL(NCOMP) + EHIT
 15   CONTINUE
      GOTO 10
*
      EDEP = 0.
      EDEP5M = 0.
      DO 25 I=1,1560
         EDEP = EDEP + EXTAL(I)
         IF (EXTAL(I).GT.5.) EDEP5M=EDEP5M+EXTAL(I)
 25   CONTINUE
*
* Fill histograms:
      IF (EDEP.GT.0.)   CALL HF1(1,EDEP,  1.)
      IF (EDEP5M.GT.0.) CALL HF1(2,EDEP5M,1.)
      RETURN
*
 950  CONTINUE
      CALL ERRMSG(0,'MYEVNT: ILLEGAL PARAMETERS; REJECT EVENT')
      RETURN
      END
```

```
      SUBROUTINE MYLAST
+SEQ,GCKINE.
+SEQ,QUEST.
*
* Create and print histograms:
      CALL HISTDO
*
      RETURN
      END
```

6.3 Output

By default, data is recorded in exchange tape format. Even when one is writing to disk, one should normally use this format, since it allows one to later transfer the data to tape and to transport it to another machine. Disk format uses extremely short blocks and is not efficient to use on tapes. Native format prevents the transport of the data to other machine types.

6.3.1 Data Banks

The data banks produced are selected with the `RAWD` data card and are usually `EVHD`, `RPWC`, `RJDC`, `RBCL`, `RBCF` and `RMCB`. The meanings of the first five are as usual (event header, PWC, JDC, 2282 and FERA). One should usually record all of these banks. `RMCB` must always be selected to pass certain parameters to the analysis code; these informational banks are discussed in the next section.

6.3.2 Informational Banks

CBGEANT can produce several informational banks which make the running conditions of the Monte Carlo an integral part of the output data. These allow suitable settings to be chosen by the data analysis programs, including chamber rotations, suitable drift tables, whether or not to add further detector noise, etc. The format of the top-level informational bank `MCIN` is described in Table 4. Several other banks are linked to `MCIN` in the order listed:

`PHYS`	CBGEANT physics options and cuts: see Table 5
`KINE`	Kinematic parameters (usually from the `KINE` data card): see Table 6
`VERT`	Annihilation vertex distribution: see Table 7
`GEOM`	Geometry and digitization parameters: see Table 8
`SWIT`	Switches (usually from the `SWIT` data card): see Table 8
`DIST`	Applied experimental distortions: see Table 10

Offset	TYPE	*Quantity*
+1	HOLLERITH	Machine type for which code was extracted
+2	INTEGER	Random seed #1
+3	INTEGER	Random seed #2
+4	REAL	CBGEANT version number
+5	INTEGER	CBGEANT update number
+6	HOLLERITH	JDC patch used: 4HFJDC (`FASTJDC`) or 4HPJDC (physical)
+7	REAL	GEANT version number

Table 4: Data stored in the `MCIN` bank

6.4 Interactive Mode

The interactive mode of GEANT is quite useful; it is documented in the `XINT` section of the GEANT manual. For event simulation (as opposed to detector development) the most commonly used commands include:

HELP	Help menus
TRIGGER *N*	Start *N* events
SWITCH *ISWITCH IVAL*	Change switch *ISWITCH* to *IVAL*
DEBUG *IDEB*	'ON ': enable debugging; 'OFF': disable debugging
DUVIEW *NAME TYPE CPXTYP*	Plot a predefined detector view
DXYZ	Plot tracks (fill `JXYZ` by setting switch 10)
NEXT	Clear graphics display
EXIT	Leave GEANT

The only command needing explanation is `DUVIEW` since its arguments are defined in CBGEANT, rather than GEANT. A number of views are pre-defined, with the detector being shown cut on a plane at or near the center. (The cut plane is sometimes offset when elements such as crystals would otherwise not be seen.) Because tracks are projected onto the display plane, one must sometimes examine all three views to see what has really happened. The defined options are:

	Detector	View	Complexity
`DUVIEW`	`PW` (PWC)	`TOP`	`FULL` (Detailed)
	`GH` (GHT)	`SIDE`	`FAST` (Simplified)
	`JD` (JDC)	`FRON`	
	`BC` (Calorimeter)		
	`CB` (Overall)		

Offset	TYPE	*Quantity*
+1	INTEGER	`IANNI`
+2	INTEGER	`IBREM`
+3	INTEGER	`ICOMP`
+4	INTEGER	`IDRAY`
+5	INTEGER	`IDCAY`
+6	INTEGER	`IHADR`
+7	INTEGER	`ILOSS`
+8	INTEGER	`IMULS`
+9	INTEGER	`IMUNU`
+10	INTEGER	`IPAIR`
+11	INTEGER	`IPFIS`
+12	INTEGER	`IPHOT`
+13	INTEGER	`IRAYL`
+14	REAL	`CUTGAM`
+15	REAL	`CUTELE`
+16	REAL	`CUTNEU`
+17	REAL	`CUTHAD`
+18	REAL	`CUTMUO`
+19	REAL	`BCUTE`
+20	REAL	`BCUTM`
+21	REAL	`DCUTE`
+22	REAL	`DCUTM`
+23	REAL	`PPCUTM`
+24	REAL	`TOFMAX`
+25	REAL	`GCUTS(1)`
+26	REAL	`GCUTS(2)`
+27	REAL	`GCUTS(3)`
+28	REAL	`GCUTS(4)`
+29	REAL	`GCUTS(5)`
+30	REAL	Tracking limit on $\sqrt{x^2+y^2}$
+31	INTEGER	1: Fixed energy cuts 0: Variable energy cuts
+32	INTEGER	Number of particle types to reject
+33 – +42	INTEGER	List of particle types to reject
+43	INTEGER	Number of vertex X rays

Table 5: Data stored in the `PHYS` bank

Offset	TYPE	*Quantity*
+1	INTEGER	`IKINE` (Event generator used)
+2	REAL	`PKINE( 1)`
+3	REAL	`PKINE( 2)`
+4	REAL	`PKINE( 3)`
+5	REAL	`PKINE( 4)`
+6	REAL	`PKINE( 5)`
+7	REAL	`PKINE( 6)`
+8	REAL	`PKINE( 7)`
+9	REAL	`PKINE( 8)`
+10	REAL	`PKINE( 9)`
+11	REAL	`PKINE(10)`

Table 6: Data stored in the `KINE` bank

Offset	TYPE	*Quantity*
+1	INTEGER	Vertex distribution type
+2	REAL	$\overline{x}$ of uncut vertex dist.
+3	REAL	$\overline{y}$ of uncut vertex dist.
+4	REAL	$\overline{z}$ of uncut vertex dist.
+5	REAL	σ_x of vertex dist.
+6	REAL	σ_y of vertex dist.
+7	REAL	σ_z of vertex dist.
+8	REAL	Upper limit on $\sqrt{x^2 + y^2}$ of vertices
+9	REAL	Lower limit on z of vertices
+10	REAL	Upper limit on z of vertices

Table 7: Data stored in the `VERT` bank

Offset	TYPE	*Quantity*
+1	REAL	Magnetic field at origin (kilogauss)
+2	INTEGER	GEANT tracking method in PWC/JDC region
+3	INTEGER	Bit mask of `GEOM`, `PREC`, `SETS` and `DIGI` data
+4	INTEGER	Liquid target: 1 for 1989–1990, 2 for 1991–
+5	INTEGER	1=LH$_2$, 2=LD$_2$ as liquid target filling
+6	REAL	Z offset of target position
+7	REAL	Z position of downstream veto counter
+8	REAL	Energy cut on veto counter
+9	INTEGER	Design number of PWC #1
+10	INTEGER	Design number of PWC #2
+11	REAL	Rotation angle of PWC #1 (CBGEANT)
+12	REAL	Rotation angle of PWC #2 (CBGEANT)
+13	INTEGER	Design number of GHT
+14	REAL	Rotation angle of GHT
+15	INTEGER	Design number of JDC
+16	REAL	Rotation angle of JDC (CBGEANT)
+17	REAL	Lorentz angle used for JDC
+18	REAL	B-field for JDC gas file (kilogauss)
+19	REAL	Pressure for JDC gas file (Torr)
+20	INTEGER	JDC gas file: $10000 \times (\%\ CO_2) + 10 \times (\%\ C_4H_{10})$
+21	INTEGER	Date electrostatic solution first used (DDMMYY)
+22	INTEGER	Version number of JDC gas file
+23	INTEGER	BC copper cooling collars: 1 if used, 0 otherwise
+24	INTEGER	Bit mask encoding the `RAWD` banks stored

Table 8: Data stored in the `GEOM` bank

Offset	TYPE	*Quantity*
+1	INTEGER	Switch 1
+2	INTEGER	Switch 2
+3	INTEGER	Switch 3
+4	INTEGER	Switch 4
+5	INTEGER	Switch 5
+6	INTEGER	Switch 6
+7	INTEGER	Switch 7
+8	INTEGER	Switch 8
+9	INTEGER	Switch 9
+10	INTEGER	Switch 10

Table 9: Data stored in the `SWIT` bank

Offset	TYPE	*Quantity*
+1	REAL	PWC1: Efficiency
+2	REAL	PWC1: Probability hit gives adjacent hit
+3	REAL	PWC1: Probability allowed noise hit occurs
+4	INTEGER	PWC1: Allowed number of noise hits
+5	REAL	PWC2: Efficiency
+6	REAL	PWC2: Probability hit gives adjacent hit
+7	REAL	PWC2: Probability allowed noise hit occurs
+8	INTEGER	PWC2: Allowed number of noise hits
+9	INTEGER	Variation in crystal calib. constants: -1: table; 0: none; +1: Gaussian
+10	INTEGER	FERA noise (coherent): -1: table; 0: none; +1: Gaussian
+11	INTEGER	FERA noise (incoherent): -1: table; 0: none; +1: Gaussian
+12	INTEGER	2282 noise (coherent): -1: table; 0: none; +1: Gaussian
+13	INTEGER	2282 noise (incoherent): -1: table; 0: none; +1: Gaussian
+14	REAL	1σ variation in crystal calibration
+15	REAL	1σ coherent FERA noise
+16	REAL	1σ incoherent FERA noise
+17	REAL	1σ coherent 2282 noise
+18	REAL	1σ incoherent 2282 noise
+19	REAL	JDC signal random variation
+20	REAL	JDC signal noise
+21	REAL	JDC signal spreading in X (cm)
+22	REAL	JDC signal spreading in Y (cm)
+23	REAL	JDC resolution in Z (cm)
+24	REAL	Limit on JDC Z resolution tails (cm)
+25	INTEGER	JDC signal threshold
+26	REAL	Normalization of JDC signal height
+27	REAL	Width of avalanche distribution
+28	REAL	Probability of double JDC pulses
+29	REAL	Probability of triple JDC pulses
+30	REAL	JDC signal wire electrical half length

Table 10: Data stored in the `DIST` bank

7 Shortcomings

7.1 GEANT

An incomplete list of notes on physics in GEANT:

- According to the GEANT manual (section `CON110`), radiation lengths are calculated with the EGS3 algorithm - *not* the Tsai algorithm used in EGS4. Thus, it is not particularly accurate.
- Bremsstrahlung in the field of electrons is treated as a scaling factor to the nuclear bremsstrahlung, even though the differential cross section has a different shape in reality. Further (as in EGS4), only the average angle is assigned to the emitted γ's (tending to overly restrict the angular development of showers).
- Like other Monte Carlos, radiative corrections to various other processes are usually ignored. The error can be several per cent. An example would be 3γ annihilation in high Z materials.
- There is no calculation of the transverse displacement of particles during a step due to multiple scattering.
- There is little detailed treatment of atomic shell effects, or of other effects below energies of several KeV. This complicates the making of detailed simulations of thin, sensitive regions such as drift chamber cells.

7.2 CBGEANT

7.2.1 Event Initialization

- The branching ratios in the BIGBANG event generator could be improved further, and additional channels defined, especially for strange particles.
- Additional particles should be defined, including the K^* group.
- Event triggers have not been implemented.
- There is no provision for user access to the data in an event until it has completed. In certain specialized cases, it may be that the user will want access to information earier, so as to allow early rejection of the event. The obvious place for such access is in GUSTEP. However, such a change should take into account that this routine is called very frequently, and that the overhead needed to make a subroutine call to even a dummy routine could be noticable.

7.2.2 Target Area

- Although geometry is defined for a high pressure target, it is more or less arbitrary, since no detailed design exists. The default pressure is also arbitrary. A real design should be coded if someone intends to actually use it.

- The silicon beam counters have not been included.
- While the so-called "π" veto counter has been included, is has not yet been set up to either reject the event or to record the energy deposition in a data bank.

7.2.3 PWC/GHT

- Digitization for the GHT has not yet been implemented.
- Details such as beam windows are missing from the GHT description.
- The efficiency applied in the PWC digitization does not consider the probable correlation with dE/dx and other factors.

7.2.4 JDC

- Some effects, such as amplifier input impedance, due to the JDC electronics are not handled.
- There is no provision for resolution degradation due to systematic effects such as wire positioning inaccuracy or dead wires. The effect of this is somewhat different than that due to random factors.
- The Z resolution is fixed at 0.75 cm and does not reflect the variation with pulseheight and position that one would expect to be present.
- The `FASTJDC` algorithm uses a straightline to connect the entrance and exit points of a particle in a drift cell. This approximation begins to break down for low-momentum particles.

7.2.5 Barrel Calorimeter

- The amount of light detected by the photodiode for a given energy deposition in the wavelength shifter, compared to a crystal, is not known accurately. A measurement has yet to be made.
- No correction is made for variation in the light collection with position within a crystal. This correction is difficult, because it varies significantly from one crystal to the next and because the existing measurements used photomultiplier tubes, rather than photodiodes, and these do not give compatible results.
- No pedestal-suppressed mode is yet supported.
- Treatment of certain variables related to inorganic scintillators (`GHCOR*`) is somewhat uncertain. An adequate answer from F.Carminati has not been forthcoming.

7.2.6 Miscellaneous

- CBGEANT has not yet been split so as to separate the simulation of the physics from the simulation of detector imperfections (with intermediate files being produced to connect the two parts).
- There is no formal support in the code for the IBM RISC/6000 or NeXT processors, which are likely to be used by members of the collaboration and are now formally supported by CERN. The CDC 4600-series workstation is supported only in so far as it is compatible with the DECstation.
- The Fortran code is still not strictly ANSI-77 compliant, despite improvements in that direction.
- There have not yet been detailed comparisons of Monte Carlo results with experiment. It is certain that various parameters will need to be tuned, and likely that some algorithms will need to be improved.
- The magnetic field inhomogeneity implemented in the GUFLD subroutine is probably not accurate; more information is needed.

8 Analysis of CBGEANT Output

For the next stage of the analysis (e.g., CBDISP), LOCATER has to know how to allign the JDC and PWC with the crystals. To analyze data from the default Monte Carlo, the JDC sector 1 sense plane must be placed 0.79° past the $+Y$ axis (90.79° past the $+X$ axis) and the PWC1 and PWC2 wires zero must be placed on the $+Y$ axis ($\pi/2$ radians from the $+X$ axis). The angle conventions are different between GEANT and LOCATER; GEANT angles refer to the edge of a JDC sector and the plane halfway between PWC wires, while LOCATER alignments are to the sense plane or wire.

For CBGEANT 4.03–4.05, the allignment was done with data cards, but it is now done automatically by passing the angles in the `GEOM` bank, which are then used automatically by LOCATER—provided one has a version more recent than 10-FEB-1992. In the event that you have an older version of cbg or LOCATER, or if you failed to select the `RAWD` data card option `RMCB`, the data cards to use at analysis time are:

```
ANGL 90.79 1.0
OPWC 1.570796 1.570796
```

Similarly, an obsolete version of LOCATER may need a JDC gain table for the Monte Carlo. This should show all wire half lengths to be 23.68 cm, not 20.00 cm or something else. Because of the change in default wire length, a different table would be needed than the one used for CBGEANT 4.05 and earlier. If you are uncertain about which one you have, copy `JDC_GAIN_MC406.TBL` from `VSXTAL::[CB.GEORGE]JDC_GAIN_MC406.TBL`.