CLAS-NOTE 95-008

CLAS Geometry Format

Version 1.00 (April 3, 1995)

Etienne Burtin, Dieter Cords, Dave Heddle, Kin Leung, Bogdan Niczyporuk

1.0 Introduction

The geometry of the CLAS detector is complex. In this document we define the geometry parameters for simulation, reconstruction and display. The objective is to have a very basic set of parameters to describe the location and orientation of each detector component and thus to record any shifts due to misalignment.

The geometry information is made available to all CLAS software in the form of BOS banks which contain binary numbers. The BOS banks themselves can be converted to ASCII arrays, in order for their contents to be viewed. Such a user interface would also make it possible to enter the initial survey data for the various detector components.

In this note we define the contents of the BOS banks, describe the user interface and the corresponding ASCII arrays, and finally give the convention for maintaining various versions of geometry records.

2.0 Geometry BOS Banks

The objective is to supply enough information to describe the active volumes of the detector components. In particular, one wants to be able to derive the positions of the sensitive elements - that is the positions of the drift chamber wires as well as the entrance surfaces for Cerenkov, scintillation counter and electromagnetic calorimeter elements - from this basic set of geometry parameters. In most cases it will be convenient to use a subroutine to obtain the location and orientation of the detector elements. In fact, for the drift chamber it will be mandatory to use a subroutine to get the wire positions.

The geometry banks contain a sufficient amount of information for reconstruction, analysis and event display. At this stage it is not attempted to also define the material types and thicknesses needed in addition for simulations.

For the description of the detector elements we follow the same numbering scheme that was used for the CLAS Event Format - see Clas-Note 94-012. In fact, the detector element IDs used below, are identical to the ones in that Clas-note. The numeric values of the parameters are given as integer and floating point numbers in Cartesian or Polar coordinates of the CLAS Laboratory Frame. All parameters, starting with I or N in their names, are 4-byte integers. All others are floating point numbers. The units are degrees for angles and cm for distances.

The naming of the geometry banks also follows the conventions in Clas-Note 94-012 by appending a G to the raw event bank names. Therefore, the basic banks defined here, are DCG, CCG, C1G, SCG, ECG and E1G for the drift chambers, the Cerenkov and scintillation counters, as well as the calorimeters. The bank numbers run from 1 to 6 for the six sectors. Normally one has to have a geometry Bos bank for each detector component in each sector. However, for some detector components only an approximate knowledge of the position is required - like for the Cerenkov counter. In this case the banks for sectors n=2..6 can be omitted, because they can be obtained from CCG 1 by rotations of (n-1)x60 degrees around the Z axis.

2.1 Drift Chamber Geometry Banks DCG

The drift chambers are the most complex detector components. Due to the large number of wires and more subtle effects - like wire sagging - it is necessary to obtain the actual position of individual wires analytically by subroutine calls. The calling sequence of the standard geometry routines is given in the next subsection.

In this section we define the format of the DCG banks which supply the geometric input parameters for those geometry routines. Like all other experimental data, the parameters are given in tabular form. Each row has NCOL=16 elements which essentially define the positions of the first and last wires for each drift chamber layer at the midplane. The number of rows is at most 34 (the number of layers) and can be reduced to 6 (one layer for each superlayer). Because of their regular structures one only needs the 3rd layer in a superlayer to obtain the wire positions for all layers by subroutine calls. The parameters of each drift chamber layer are described below:

DCG: IL NF NL X_B Y_B Z_B R Θ_{FS} Φ_{FS} Θ_{LS} Φ_{LS} Θ_{TR} DR CL SA SM

with IL = Layer number: 1...36 (layers 5 & 6 do not exist)
NF = Number of First sense wire - chosen such that wires with the same number in successive layers line up radially in one superlayer (Clas-Note 94-012)
NL = Number of Last sense wire
X_B Y_B Z_B = Coordinates of "Bogdan" center
R = Radius of sense wire plane from "Bogdan" center
Θ_{FS} = Polar angle of First Sense wire (includes displacement of mini-stagger)
Φ_{FS} = Azimuth of midplane at First Sense wire
(= 0 at ideal location in sector 1)
Θ_{LS} = Polar angle of Last Sense wire
Φ_{LS} = Azimuth of midplane at Last Sense wire
(= 0 at ideal location in sector 1)
Θ_{TR} = Angle at TRansition from cylinder to plane part in first two superlayers
(= 0 if only cylinder)
DR = Difference in Radius for neighboring layers in a superlayer
CL = Constant for calculating the wire Length
SA = Stereo Angle
SM = Size of Mini-stagger

The polar and azimuth angles, Θ and Φ, of the first and last sense wires are measured in a Lab Coordinate System displaced by X_B Y_B Z_B from the "target" position. The reconstruction of tracks is conveniently performed in Sector Coordinate Systems which can be defined by the "ideal" sector midplanes at $\Phi = 0^o, 60^o, 120^o$... (aligned along Z) <u>or</u> by the "actual" midplanes of any of the three regions. In either case, transformations between the actual and ideal systems are needed and can be approximately achieved by two angles. First one rotates by α around Z to align the X axes and then one rotates by β around X to align the Z axes:

$$\alpha = \frac{\Phi_{LS} \bullet \left(\frac{\pi}{2} - \Theta_{FS}\right) + \Phi_{FS} \bullet \left(\Theta_{LS} - \frac{\pi}{2}\right)}{\Theta_{LS} - \Theta_{FS}}$$

$$\beta = \tan\Theta_{FS} \bullet \sin\left(\Phi_{FS} - \alpha\right) \quad \text{for small} \quad \beta = \tan\beta$$

Transformations like these are best performed for small angular deviations form the ideal sector midplane, not for rotations between sectors.

Geometry parameters, needed for reconstruction in a sector framework, have to be transformed to that framework. This new set of parameters is best kept in a derived set of banks for the sector system, DCGS, which can be created at the beginning of a reconstruction run. However, it is important to maintain and update a master-set of geometry banks. This master-set is described in this document.

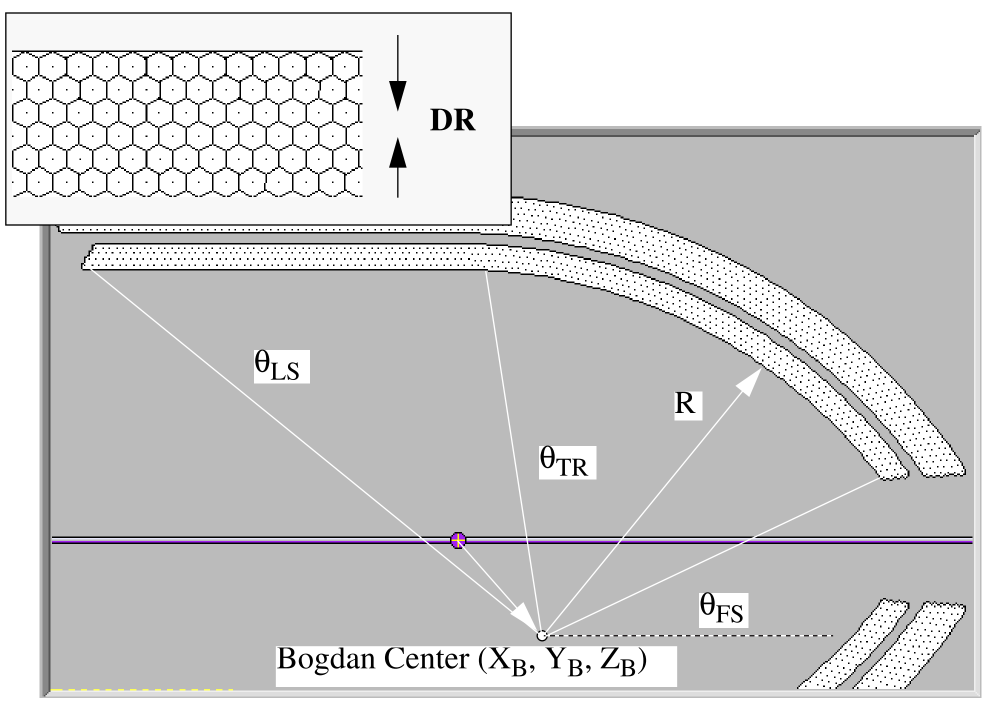


FIGURE 1. Some of the parameters for the Drift chamber bank.

2.1.1 Drift Chamber Geometry Routine

2.2 Forward Cerenkov Counter Geometry Banks CCG

The Cerenkov counters are designed as projective volumes with the origin of the Lab system as the vanishing point. The entrance window of each double unit (2 units adjacent in Φ) is perpendicular to a vector from the Lab origin which pierces the window at its center where the two units meet. The objective is to define the entrance window by a center-line connecting the middle of the outer edges - X_L Y_L Z_L for low Φ (ID odd) and X_H Y_H Z_H for high Φ (ID even) - and its width W:

CCG: ID X_L Y_L Z_L X_H Y_H Z_H W_{IN} D

with ID = 1,3,5,..35 (see Clas-Note 94-012) for units at low Φ
X_L Y_L Z_L = Coordinate at middle of low Φ end of unit with odd ID
X_H Y_H Z_H = Coordinate at middle of high Φ end of unit with even ID
W_{IN} = Inner Width
D = Depth/Thickness (along the projection line, i.e. the ray from the origin)

Each row has NCOL=9 entries and there are up to 18 rows. It could be sufficient to enter a fewer number of rows (e.g. every 3rd double unit) and to interpolate between the missing ones.

The vanishing point of the Cerenkov elements may not completely agree with the origin of the Lab System (due to misalignment). However, such a deviation will not be recorded here because a precise knowledge of the orientation of the gas volume is not required.

2.3 Large Angle Cerenkov Counter Geometry Banks C1G

The Cerenkov counters away from the forward direction are at polar angles larger than 45^o. The geometry of the large angle counter adjacent to 45^o is given in Bos bank C1G. If there is a 3rd counter at even larger angles it will be described by C2G.

The format of this bank is the same as for CCG.

2.4 Scintillation Counter Geometry Banks SCG

For the scintillation counters we define the front or inner surfaces of each counter by a center-line, connecting the middle of the outer edges of each paddle, and the paddle width. The scintillation counters are a flat structure, and the locations of the outer edges are given in Cartesian coordinates:

SCG: ID IP X_L Y_L Z_L X_H Y_H Z_H W T

with ID = 1...48 = ID for each pair of electronics channels (see Clas-Note 94-012)
IP = 1...6 = Id of Panels which differ in orientation or in paddle width

X_L Y_L Z_L = Coordinates at center of Low Φ (left) end of paddle
X_H Y_H Z_H = Coordinates at center of High Φ (right) end of paddle
W = Width of paddle
T = Thickness of paddle

Each row in this geometry table consists of NCOL=10 elements. From ID=40 onwards two adjacent paddles (of equal widths) are feeding into one pair of electronics channels, and, therefore, they are counted as one: W is the combined width; and the effective length of the center-line is the average length of the two paddles. The actual length of each individual paddle can be obtained by combining the data of two adjacent units in the same panel. Also the slant (orientation) along the width can be inferred from two neighboring paddles.

The two most forward planes are panels 1 and 2. They are composed of 23 and 11 paddles respectively. The two backward planes are subdivided into two panels each because the paddle width changes. These panels are numbered 3 to 6 and have 5, 3, 4, 2 paddles each. The geometry table can consist of up to 48 rows, but it is sufficient to only have 12 entries for the first and last paddle positions for each panel.

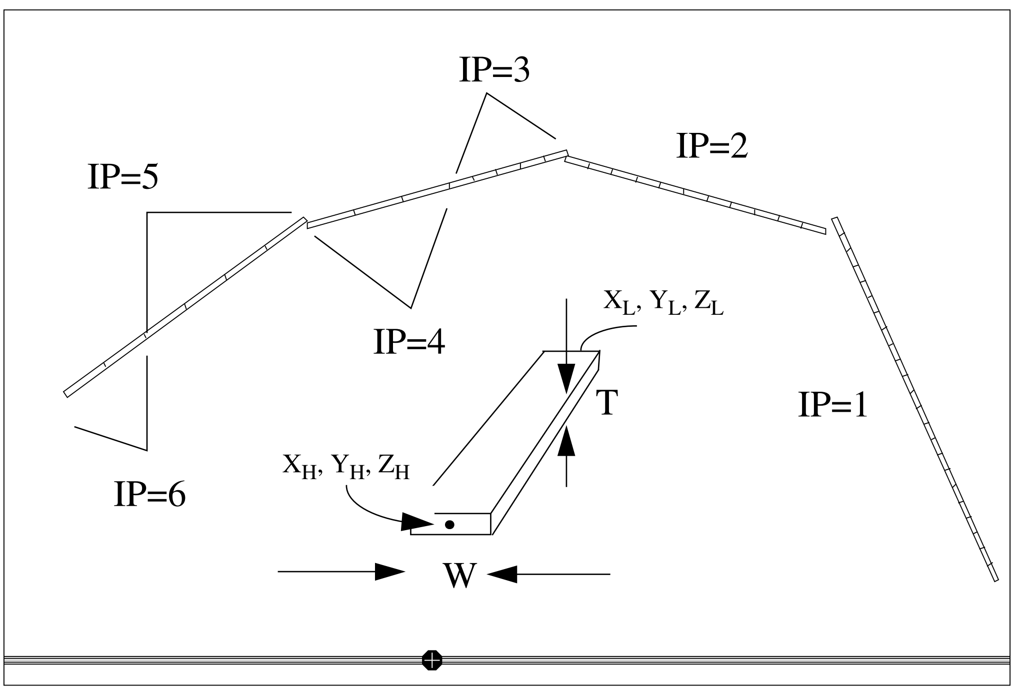


FIGURE 2. Some parameters from the scintillator bank.

2.5 Forward Electromagnetic Calorimeter Geometry Banks ECG

The forward calorimeters have scintillator slabs (interspersed with lead sheets) in three different directions (u, v, w strips). In a projective geometry, several successive slabs are ganged together. In particular, 5 successive slabs in each of the u, v, w directions are ganged together for the inner calorimeter, and 8 successive u, v, w slabs are ganged together in the outer calorimeter. Therefore, there are 6 effective layers and 36 strips each of the u, v, w directions. The objective, here, again is to define the entrance window of each stack of slabs and the depth of each stack.

The entrance window (front surface) of each strip (stack of slabs) has a trapezoidal shape so that all 36 strips of any of the 6 layers - joined together - form a triangle. With each strip identifier IS one stores the corner coordinates of the longest edge of the entrance window trapezoid. One then obtains all four corners of that trapezoid in conjunction with the next lower strip identifier IS-1. The non-physical IS=0 describes the tip of the triangle ($\mathbf{X}_L = \mathbf{X}_H$) and the largest IS=36 the baseline of the triangle. The complete ID combines the layer and strip numbers, IL and IS, as shown in the following table:

ECG: ID X_L Y_L Z_L X_H Y_H Z_H W D

with ID = 256*IL + IS for IL = 1...6 layers and IS = 0,1...36 strips
X_L Y_L Z_L = Corner Coordinates at Low Θ / Φ end of longest trapezoidal side
X_H Y_H Z_H = Corner Coordinates at High Θ / Φ end of longest trapezoidal side
W = Width of trapezoid
D = Depth of stack of slabs (along the projection line)

Each row in this table consists of NCOL=9 elements and there are a minimum of 2*6 rows - if only strips 0 and 36 for each of the 6 layers are given - or a maximum 37*6 rows.

2.6 Large Angle Electrom. Calorimeter Geometry Banks E1G

As one moves away from the forward calorimeters to larger polar angles, the detector assembly may have one or two more calorimeters, the geometry banks of which are called E1G for the first (or closest) one and possibly E2G. These large angle calorimeters are constructed differently from the forward calorimeters. They have slabs or strips in only two directions (x and y). There are 8 & 7 radially successive slabs in the respective x & y directions ganged together for the inner calorimeter; and 9 successive x as well as y slabs are combined for the outer calorimeter. Therefore, there are effectively 4 sensitive layers. The 40 x strips are parallel to Z in the Lab Frame and numbered as ϕ increases. The 24 y strips are perpendicular to Z and numbered with increasing Θ. As before, we define the now rectangular entrance window of each strip by the four corner coordinates identified by IS and IS-1. The non-physical IS=0 describes the lower edge and the largest IS (40 or 24) the upper edge of the entire layer surface.

ECG: ID X_L Y_L Z_L X_H Y_H Z_H W D

with ID = 256*IL + IS for IL=1..4 layers; IS=0..40 for x (odd L) & 0..24 for y (even L)
X_L Y_L Z_L = Corner Coordinates at Low Θ / Φ end of upper rectangle side
X_H Y_H Z_H = Corner Coordinates at High Θ / Φ end of upper rectangle side

W = Width of rectangle
D = Depth of stack of slabs (along projection line)

Each row in this table consists of NCOL=9 elements and there are a minimum of 2*4 rows - if only the zeroth and last strips of the 4 layers are given - or a maximum of 2*41+2*25 rows.

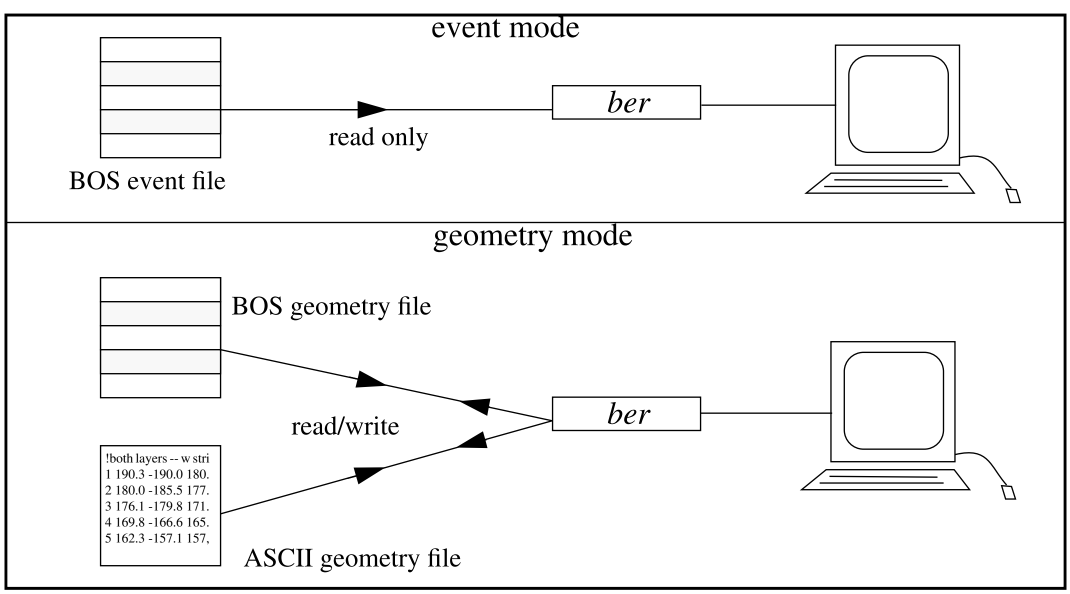
3.0 User Interface

3.1 Introduction

The command-line based user interface into the geometry database is through the program *ber* [1] (bOS eVENT rEADER). This program was originally written to be (and is still used as) a reader/debugging tool for CLAS events in BOS format. When it was decided that the geometery data would also be stored in BOS format, it seemed natural to generalize *ber* rather than write a completely new application.

By design, the marriage of the event bank and geometry bank capabilities of *ber* was <u>not</u> integrated and seemless. One uses *ber* in either geometry mode or event mode, as described below. This is done both to avoid confusion and in recognition that the operations one may want to perform with the geometry banks are different from those performed on the event banks. In particular, the ability to modify banks is needed in the geometry mode but not in the event mode. Also, the geometry mode can be driven (at least for now) from an ASCII geometry description as well as a file of BOS banks.

Schematically, then:



1. *ber* should be referred to by its spelling, “b-e-r”, not as if it were a word pronounced *bûr*.

FIGURE 3. Schematic representation of the functionality of *ber*.

When new banks are "sanctified" by the CLAS Software Group, *ber* is then trained to recognize and display them in ASCII tabular form.

3.2 *ber* Commands

ber has the following usage:

```
$ ber [{-v, -g}] [filename]
```

the flags **v** and **g** are optional *and exclusive*: **v** signifies that you want events to be displayed in "verbose" style (described below) and thus implies that you are in event mode. The **g** flag signifies that you want to run *ber* in geometry mode.

The optional **filename** refers to either a file of BOS events (in the case of event mode), a file of BOS geometry banks, or the ASCII geometry description file. Nonsensical commands like

```
$ ber -g myevents.evt
```

while not cause *ber* to crash, but might as well since no useful output can ensue;[2] *ber* will, because of the **-g**, be looking for geometry banks and ignoring event banks.

3.2.1 Geometry Mode

If *ber* is launched in geometry mode with the command

```
$ ber -g
```

the following menu is presented:

```
Possible Commands (first char sufficient):

        d*isplay a memory resident bank
        e*nglish description of a bank
        f*lush memory resident banks
        m*odify a memory resident bank
        q*uit ber
        r*ead a geometry file
        s*ummarize memory resident banks
        w*rite a geometry file
GEO >
```

As the header informs, the first letter of each command is sufficient (and necessary). The prompt **GEO >** is a constant reminder that you are in geometry mode. We now proceed through the possible commands.

2. Unless, of course, you have inexplicably named a geometry file "myevents.evt".

3.2.1.1 d*isplay

Displays the contents of a bank:

```
GEO> d
Which Bank? (enter the number)
1       Drift Chamber        "DCG "
2       Scintillators        "SCG "
3       Forward EC           "ECG "
4       Large Angle EC       "E1G "
5       Forward Cerenkovs    "CCG "
6       Large Angle Cerenkovs "C1G "

BANK > 1
SECTOR [1-6] > 1
LAYER  [1-4, 7-36] > 7

-----------------------------------
Present values in bank: "DCG "
Sector 1
 IL          7              Layer number.
 NF          1              ID of first sense wire.
 NL          192            ID of last sense wire.
 XB         -67.6      cm     X coordinate of "Bogdan" center.
 YB          0.0       cm     Y coordinate of "Bogdan" center.
 ZB          29.3      cm     Z coordinate of "Bogdan" center.
 R          235.26     cm     Radius of sense wire plane "Bogdan" center.
 ThetaFS     23.9      deg    Polar angle of first SENSE wire.
 PhiFS        0.0      deg    Azimuthal angle of first SENSE wire.
 ThetaLS    134.3      deg    Polar angle of last SENSE wire.
 PhiLS        0.0      deg    Azimuthal angle of last SENSE wire.
 ThetaTR     90.0      deg    Transistion from cylidrical to planar.
 StereoAng    6.0      deg    The stereo angle.

Display another bank? [y/n] >
```

3.2.1.2 e*nglish

Provides an english description of a bank format. For example:

```
GEO> e
Which Bank? (enter the number)
1       Drift Chamber        "DCG "
2       Scintillators        "SCG "
3       Forward EC           "ECG "
4       Large Angle EC       "E1G "
5       Forward Cerenkovs    "CCG "
6       Large Angle Cerenkovs "C1G "
BANK > 1

-----------------------------------
Description of bank: "DCG "
Drift Chamber Geometry
 IL        I*32           Layer number [1..36]. (Layers 5 and 6 are "virtual".)
 NF        I*32           ID of first sense wire. (not necessarily "1", c.f.
                          CLAS-Note 94-012.)
 NL        I*32           ID of last sense wire.
```

```
 XB         F*32      cm    X coordinate of "Bogdan" center.
 YB         F*32      cm    Y coordinate of "Bogdan" center.
 ZB         F*32      cm    Z coordinate of "Bogdan" center.
 R          F*32      cm    Radius of sense wire plane "Bogdan" center.
 ThetaFS    F*32     deg    Polar angle of first SENSE wire (from Bogdan Center.)
 PhiFS      F*32     deg    Azimuthal angle of first SENSE wire
 ThetaLS    F*32     deg    Polar angle of last SENSE wire (from Bogdan Center.)
 PhiLS      F*32     deg    Azimuthal angle of last SENSE wire
 ThetaTR    F*32     deg    Polar ang (midplane) of the transistion from cylidrical
                           to planar geom. Relevant only for Rgn 1 (layers 1-12.)
 StereoAng  F*32     deg    The stereo angle.

Another description? [y/n] >
```

3.2.1.3 f*lush

This clears all the memory resident banks, after first asking the user if they really want to do it:

```
GEO> f
flush all banks? [y/n]: y
all banks flushed
GEO>
```

3.2.1.4 m*odify

Used to modify parameters in the banks:

```
GEO> m
Which Bank? (enter the number)
1       Drift Chamber         "DCG "
2       Scintillators         "SCG "
3       Forward EC            "ECG "
4       Large Angle EC        "E1G "
5       Forward Cerenkovs     "CCG "
6       Large Angle Cerenkovs "C1G "

BANK >1
SECTOR [1-6] > 1
LAYER  [1-4, 7-36] > 7

----------------------------------
Present values in bank: "DCG "
Sector 1
[1]  IL          7               Layer number.
[2]  NF          1               ID of first sense wire.
[3]  NL          192             ID of last sense wire.
[4]  XB        -67.6      cm     X coordinate of "Bogdan" center.
[5]  YB          0.0      cm     Y coordinate of "Bogdan" center.
[6]  ZB         29.3      cm     Z coordinate of "Bogdan" center.
[7]  R         235.26     cm     Radius of sense wire plane "Bogdan" center.
[8]  ThetaFS    23.9      deg    Polar angle of first SENSE wire.
[9]  PhiFS       0.0      deg    Azimuthal angle of first SENSE wire.
[10] ThetaLS   134.3      deg    Polar angle of last SENSE wire.
[11] PhiLS       0.0      deg    Azimuthal angle of last SENSE wire.
[12] ThetaTR    90.0      deg    Transistion from cylidrical to planar.
[13] StereoAng   6.0      deg    The stereo angle.

Which parameter to modify? ([1-13], or 0 to quit) > 7
```

```
Current value of parameter [7]: 235.2600
NEW VALUE > 233.1

-----------------------------------
Present values in bank: "DCG "
Sector 1
[1]  IL          7              Layer number.
[2]  NF          1              ID of first sense wire.
[3]  NL          192            ID of last sense wire.
[4]  XB        -67.6     cm     X coordinate of "Bogdan" center.
[5]  YB          0.0     cm     Y coordinate of "Bogdan" center.
[6]  ZB         29.3     cm     Z coordinate of "Bogdan" center.
[7]  R         233.10    cm     Radius of sense wire plane "Bogdan" center.
[8]  ThetaFS    23.9     deg    Polar angle of first SENSE wire.
[9]  PhiFS       0.0     deg    Azimuthal angle of first SENSE wire.
[10] ThetaLS   134.3     deg    Polar angle of last SENSE wire.
[11] PhiLS       0.0     deg    Azimuthal angle of last SENSE wire.
[12] ThetaTR    90.0     deg    Transistion from cylidrical to planar.
[13] StereoAng   6.0     deg    The stereo angle.

Which parameter to modify? ([1-13], or 0 to quit) > 0
Modify another bank? [y/n] > n

GEO>
```

3.2.1.5 q*uit

Quits *ber* (completely, not just the geometry mode.)

3.2.1.6 r*ead

This reads a geometry file, either ASCII or BOS, and loads the banks. *It does not flush the banks. It overwrites any banks it finds in the file and leaves other banks alone.*

3.2.1.7 s*ummarize

This displays a table of which banks are presently resident:

```
GEO> s

SECTOR     1      2      3      4      5      6
-------------------------------------------------
 DCG       *      *      *      *      *      *
 SCG       *      *      *      *      *      *
 ECG       *      *      *      *      *      *
 E1G
 CCG       *      *      *      *      *      *
 C1G

GEO>
```

3.2.1.8 w*rite

Writes a file (either ASCII or BOS) from the resident banks. If the file exists, it will ask if you want to overwrite:

```
GEO> w
ASCII or BOS? [a/b]: b
filename> geometry.bos
file /clas/clas02/usr/users/heddle/BER/geometry.bos exists.
Overwrite? [y/n]: y
file /clas/clas02/usr/users/heddle/BER/geometry.bos written.
GEO>
```

3.2.2 Event Mode

Running *ber* on a BOS event file merely spits out the FPACK summary:

```
$ ber allsect.evt
********************************************************************
*
*
*         I N P U T / O U T P U T    S T A T I S T I C S
*
*
********************************************************************
 Data name .......... BOSINPUT
 File name .......... allsect.evt
 Host name .......... ---

 Unit number ...........................................  1
 Access ................................................
  sequential
 Form ..................................................
      binary
 Action ................................................
        read
 Physical record length in bytes ...........................  131040

 Number of physical records read ..........................  1
 Number of logical records read ...........................  100

 Number of Kbytes read ....................................  102

 Average length of logical record on input ................  1020
 Number of data blocks read ...............................  683
 -------------------------------------------------------------------
```

ber becomes more useful when run with the verbose flag:

```
$ ber -v example_analyzed.evt
```

```
******************************************************************
*
*
*         I N P U T / O U T P U T     S T A T I S T I C S
*
*
******************************************************************
 Data name .......... BOSINPUT
 File name .......... example_analyzed.evt
 Host name .......... ---

 Unit number ...............................................  1
 Access ....................................................
  sequential
 Form ......................................................
      binary
 Action ....................................................
        read
 Physical record length in bytes ...........................  3600

 Number of physical records read ...........................  14
 Number of logical records read ............................  5

 Number of Kbytes read .....................................  50

 Average length of logical record on input .................  10000
 Number of data blocks read ................................  59
 -----------------------------------------------------------------

event:  3   num hits in each bank
SECT    DC      CC      CC1     SC      EC      EC1     EcPi    EcPo
  1      37       0       0       1      13       0       0      10
  2       0       0       0       0       0       0       0       0
  3       0       0       0       0       0       0       0       0
  4       0       0       0       0       0       0       0       0
  5       0       0       0       0       0       0       0       0
  6       0       0       0       0       0       0       0       0
MCIN: 1
  TG: 0
  TA: 0

===== Sector 1 =====

DC DATA   Number of Hits = 37
 layer  wire   tdc   adc
-------------------------------
     1    12    38    21
     2    11    56    65
     3    11    61    34
     4    10    86    60
     7    14    34    43
     8    13     6    25
     9    13     5    40
```

```
    10    12    28    18
    11    12    45    35
    12    11    83    23
    13    15    56    90
    14    14    41    36
    15    14   129    45
    16    13   230    40
    16    14   216     0
    17    14   120    51
    18    13    13    36
    19    14     9    44
    20    13   120    73
    21    13   230    58
    21    14   241     0
    22    13   121    63
    23    13    10    37
    24    12   104    47
    25    13   172    63
    26    12    13    97
    27    12   211    77
    28    12   286    53
    29    12    73    62
    30    11   136    97
    31    11   307   109
    32    10    82    84
    33    10   138    67
    34     9   371    90
    34    10   379     0
    35    10   152    72
    36     9    85   144

SC DATA   Number of Hits = 1
    id tdc L adc L tdc R adc R
---------------------------------------
     5   368 10180   424  8786

FORWARD ANGLE EC   Number of Hits = 13
 plane  strip   tdc   adc
-------------------------------
u-in       9   471    76
v-in      34   740    49
w-in      30   422    81
u-out      9   472   143
u-out     10   479    17
v-out     36  1402     0
v-out     35   788     9
v-out     34   745   383
v-out     33   792     2
w-out     32   438     7
w-out     31   420    95
w-out     30   416   159
w-out     29   803     5

EcPo (outer EC)  DATA   Number of Hits = 10
 id  Uid Vid Wid E(MeV) X (cm) Y (cm) Z (cm)
```

```
-----------------------------------------------------------------
   77   9  34  30 0.200  3.55 136.02 -21.95
   76   9  34  31 0.137  3.55 132.79 -16.46
   78   9  35  30 0.047  3.55 132.79 -27.43
   95  10  34  30 0.046  3.55 142.50 -21.95
   75   9  33  31 0.032  3.55 136.02 -10.97
   60   8  34  31 0.015  3.55 126.31 -16.46
   93  10  33  31 0.007  3.55 142.50 -10.97
   94  10  33  30 0.007  3.55 145.74 -16.46
   62   8  35  30 0.005  3.55 126.31 -27.43
   79   9  35  29 0.004  3.55 136.02 -32.92

MCIN DATA   Number of MCIN = 1
 Xo   Yo   Zo   Px     Py     Pz   P[GeV] M[Mev]  Q   Particle
 1.00 1.00 1.00 0.254 -0.045 0.966 1.000 139.6   -1   pi-

-------  HEAD bank --------
Version Number0
    Run Number1
  Event Number3
    Event Time0
    Event TypeGEANT
    ROC Status0
   Event Class1
  Event Weight1

Possible Commands (first char sufficient):

g*o to event
n*ext event
q*uit

BER >
```

By examination, one can see that the FPACK summary is followed by a table giving the number of hits in each bank. That is followed by a dump of each nonzero bank. Finally, the user is asked whether to go to an arbitrary event, the next event, or to quit *ber*.

3.3 Future Upgrades

The *ber* interface, being command-line driven, is ultimately unsatisfactory for use in maintaining complex banks. Wending your way through several layers of menu selections becomes very tedious. We anticipate adding a Tcl/Tk interface through the ATTIC package from FSU.

4.0 Maintaining Versions of Geometry Records

The CLAS geometry banks are kept in a logical FPACK record. Each record contains a unique and complete set of geometry BOS banks. The logical record header contains three internal identifiers or keys. The first one is a name (of up to 8 characters) which in this case is GEOMETRY. The other two identifiers are two numbers: the “first” run number

NRUNG, for which the set of geometry banks are valid, and a version control number NN. The NRUNG number is useful for keyed access. It defines a range of physics run numbers for which a particular geometry record is valid; i.e. from NRUNG in the current record up to (but not including) the next higher NRUNG in a subsequent record. For NN we will decide later whether it just caries a time stamp, e.g. the “Unix time” like in the event header, or an actual version number.

One can write several such geometry records to a single file and maintain a mini-database with keyed access. If one wants to speed up the look-up process, one can convert such a file into a “direct access” file or by employing a commercial database. This will be decided later.