Discussion of HDFast Simulations and the GlueX Portal

Rheanne Filby and Blair Jasper

SPARRO Group, Department of Physics, University of Regina

Regina, Saskatchewan, Canada, S4S 0A2

(Dated: May 2004)

Abstract

Information regarding HDFast simulations and the *GlueX* Portal are presented. The simulations that were done are specific to the Hall D barrel calorimeter (BCAL) and are therefore very useful to the SPARRO Group. A new *GlueX* Portal was installed and customized at Regina; it is written largely in PHP4 and has a MySQL database as its backend. The various components of the *GlueX* Portal and their practical uses are discussed.

Table of Contents

I. INTRODUCTION

The HDFast simulations package was employed in an attempt to determine the appropriate BCAL length needed for the *GlueX* experiment at Hall D. It allows for quick testing of changes to the geometry of the BCAL. However, due to existing problems in the HDFast software, databases needed to be altered before a thorough study of the BCAL could begin.

The *GlueX* Portal was developed to allow for better communication among the *GlueX* collaboration. However, it is not limited to a communication tool. It will be used to present pictures, documents and articles relating to the experiment. The Portal is based on the PHP-Nuke (open source) package and offers many features, including pre-installed modules and blocks. PHP-Nuke has other add-on modules available which were downloaded and installed to increase functionality. In the future, physics simulation modules and grid services will be added to the portal's complement.

II. HDFAST

A. HDFast Basics

A lot of time and effort were put into understanding, fixing and using HDFast to study the *GlueX* detector geometry. HDFast is a software package developed for integration with the Fast Monte Carlo (MCFast) simulator [1]. It depends on countless groups of modules and libraries to follow particles through the various elements of the detector and return such parameters as detector acceptance, particle momenta, Gottfried-Jackson (GJ) angles, effective mass and more. HDFast has the ability to take into account material properties, efficiencies and resolutions, and allows one to quickly see the results of making changes to the detector geometry through the use of the graphing and analysis package ROOT [2]. Where this becomes useful to SPARRO is in determining the appropriate length to build the BCAL such that acceptance is maximized. Many errors and inconsistencies had to be corrected in the HDFast package before simulations could be carried out.

B. Momenta in Root

Early on in the simulations, it was discovered that momentum was not one of the variables that we were able to view from within a ROOT browser. It was known that the momenta were being calculated and that it was possible to view the momenta for each individual event by performing a dump through a macro called *EventDisplay*. Making use of cleverly placed print out statements, the problem was traced back to an object named *ntp_maker*, which is responsible for reading in data from the root data file and writing it into an ntuple that allows us to view the data in ROOT. As it turned out, it was hard-coded into *ntp_maker* that momenta were not to be written into the ntuples unless a special flag was set upon passing the root data file to *ntp_maker*. In our case, this flag was placed within 'simulate', the script used to run every simulation (See Appendix A).

C. The Problem with the CDC

When studies of the length of the BCAL using HDFast first began, three geometry databases, HDFast.db.curtis, HDFast.db.cvs and hddsMCFast.db, existed on our computers. HDFast.db.cvs and HDFast.db.curtis were older databases which contained minor differences, whereas hddsMCFast.db was created by the hdds program that translated the geometry from XML files and will be discussed later on in the report. While HDFast.db.cvs produced understandable results, HDFast.db.curtis resulted in an unusually low acceptance ($\approx 50\%$) for reactions that involved a low energy neutral particle (see Fig. 1) Since HDFast contained the requirement that all neutral particles be detected in either the BCAL or the lead glass detector (LGD) for an event to be accepted, a thorough investigation of where these low energy particles were going was launched. Since the central drift chamber (CDC) was the only detector between the source and the BCAL, it was the obvious place to start looking.

In order to narrow in on the problem, parameters used in HDFast.db.cvs were placed into HDFast.db.curtis one by one until the acceptance was the same between the two geometry databases. The results of this task were quite unexpected. It turned out that the specific values used for r_{min} (inner radius) in the first layer of the absorber and the first two layers of anodes were causing the problem. If one, and only one, of these three parameters was changed by so much as 0.01 cm, a 40% increase in acceptance would be observed.

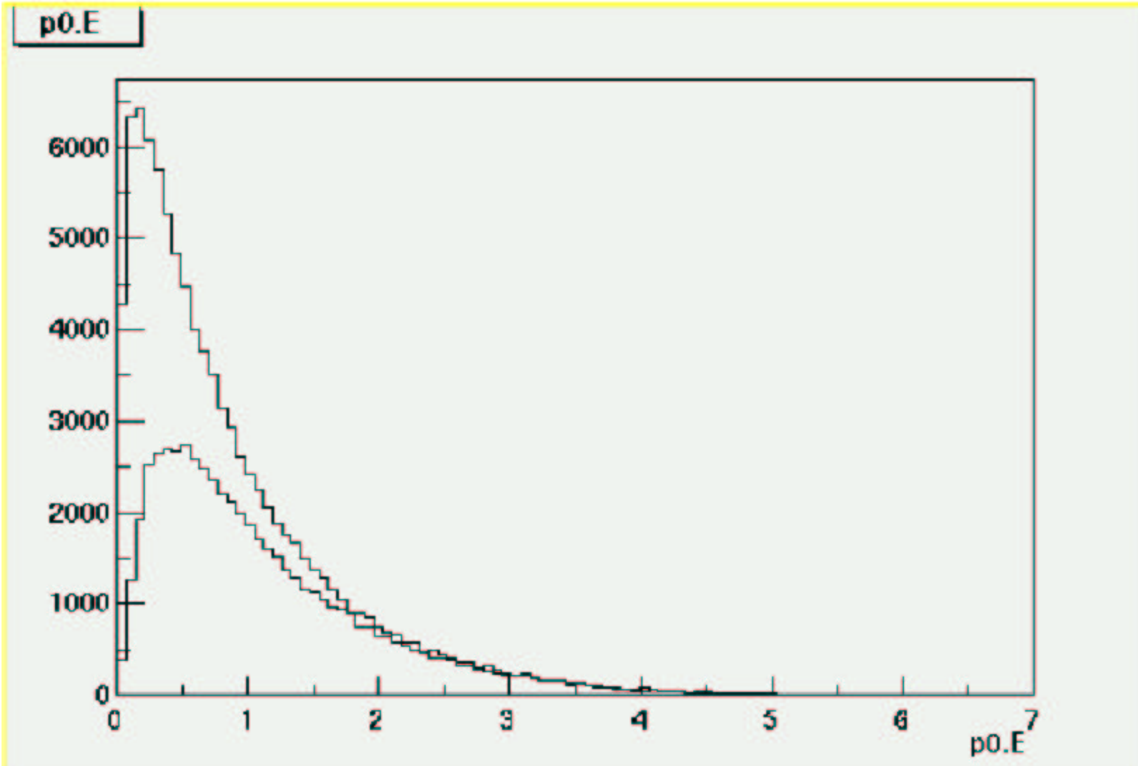

FIG. 1: *Energy distribution for a low energy photon. The lower curve is a result of a CDC database inconsistency.*

For more insight into this problem, an in-depth analysis of the effect of changing r_{min} in the absorber was performed. Simulations were run based on r_{min} values surrounding the critical point of 16.94 cm. As seen in Figure **??**, the acceptance was symmetric in both the positive and negative direction. Ultimately, no explanation for this problem was found, and this specific problem seemed to have worked itself out in the newest version of the geometry database, hddsMCFast.db.

D. XML and the Standardized Geometry Database

In the past, if the geometry of the *GlueX* detector had been changed, it was necessary to update both the HDFast and GEANT geometry databases separately. To ensure consistency between HDFast and GEANT simulations, it was decided to create a standard geometry database using Extensible Markup Language, or XML. If the geometry and mate-

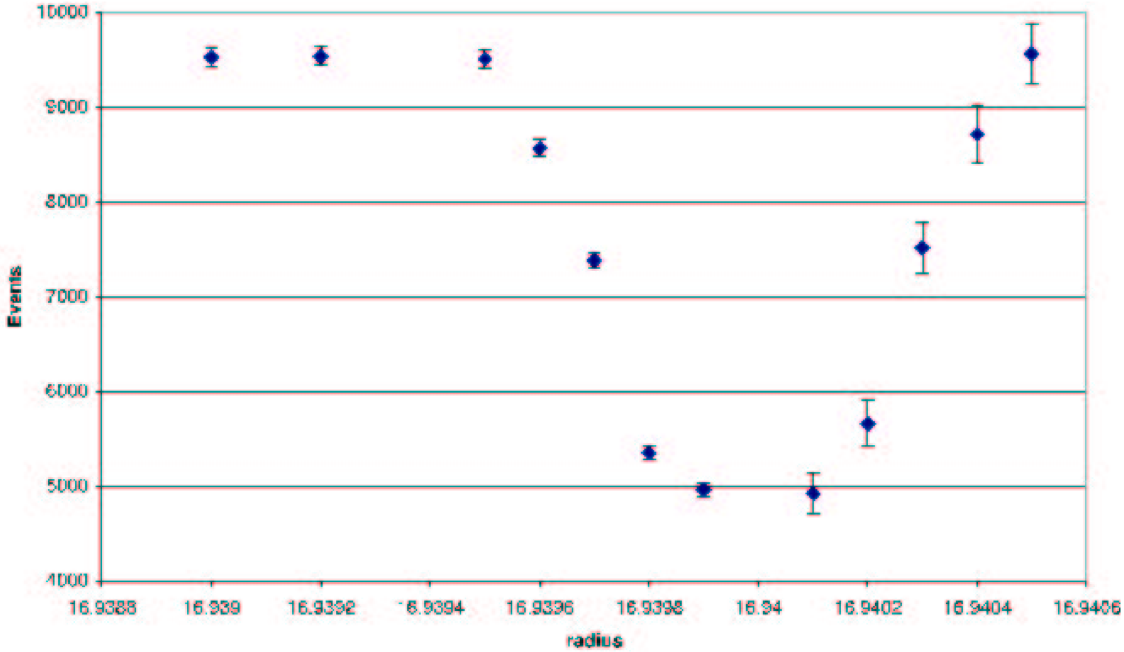

FIG. 2: *Acceptance of the detector as a function of* r_{min} *in the first layer of the absorber.*

rial definitions are all stored in XML files, it is possible to write translators that will parse this code, retrieve the necessary information and output it to a database with the proper syntax. Although this approach has much potential, major problems with the output of the database-writing program, hdds-mcfast, were discovered upon use, that have since been corrected [3].

The first and most confusing problem was that the database created with hdds-mcfast simply would cause a crash of the simulation that was running on our system (RedHat 7.3). Hours of investigation provided no further insight into the problem. Additionally, there was a lack of comments in the geometry database, making it extremely difficult to understand and use. Eventually, each geometry parameter was manually copied into an existing, working database, superdb.db, and used for the rest of the simulations.

Another problem with hdds-mcfast was the way in which materials and mixtures were handled. The syntax for a material requires a name, plus seven more parameters (A, Z, density, radiation length, nuclear interaction length, nuclear collision length and dE/dx), while a mixture requires a name, and up to eleven more parameters (the number of materials in the mixture, plus the names and relative proportions of these materials). The database only contained the last five parameters for materials that were non-elements, such as mylar.

In addition, mixtures were missing their mixing proportions. The XML files contained all of this information, indicating that hdds-mcfast was not correctly handling the XML files. To make the database usable, new material and mixture databases were created manually, using the information directly from the XML files. It should be noted that effective values for A and Z were not included in the XML files and should be added. Further, the units used in the XML files and those used in pre-hdds-mcfast databases are not consistent. The correct units required for HDFast must be determined as they can sometimes be orders of magnitude in difference.

Finally, the last problem is that the output database from hdds-mcfast does not call hitsontrack.db as other versions of the geometry database do. Without this definition, the simulation would run, however only 50% acceptance was observed. With the addition of this library, acceptance became around 90%.

E. Studying the Barrel Calorimeter

In order to test that HDFast was now working properly, a series of simulations previously reported in Hall D Note #16 [4] were repeated. The results proved to be quite comparable to those obtained in 1999, indicating that our geometry database was working as expected.

As mentioned earlier, HDFast really became useful to SPARRO when studying the acceptance of the *GlueX* detector, as a function of the length of the BCAL. Ideally, the BCAL should be at a length such that all neutral particles will hit either the BCAL or the LGD. Since HDFast had the same requirements for the acceptance of an event, it was straightforward to determine our optimum BCAL length. A series of simulations with different BCAL lengths were ran using two different decay channels: $n\eta\pi^+\pi^+\pi^-$ and $p\eta\pi^0\pi^0$. Since the $p\eta\pi^0\pi^0$ channel contained more neutral particles than $n\eta\pi^+\pi^+\pi^-$, a lower acceptance was expected, but a common optimized length for the BCAL should have been observed. As seen in Fig. 3, the acceptance of the detector plateaus for both channels at a length of approximately 440 cm.

Since the condition for an event's acceptance in HDFast matched our requirement of the *GlueX* detector, it was possible to use parameters supplied by HDFast to visually locate angles that were not being covered by the detector. Within ROOT, using the equations x=(597.5-v0.z)*p0.x/p0.z + v0.x and y=(597.5-v0.z)*p0.y/p0.z + v0.y where v0.x, v0.y

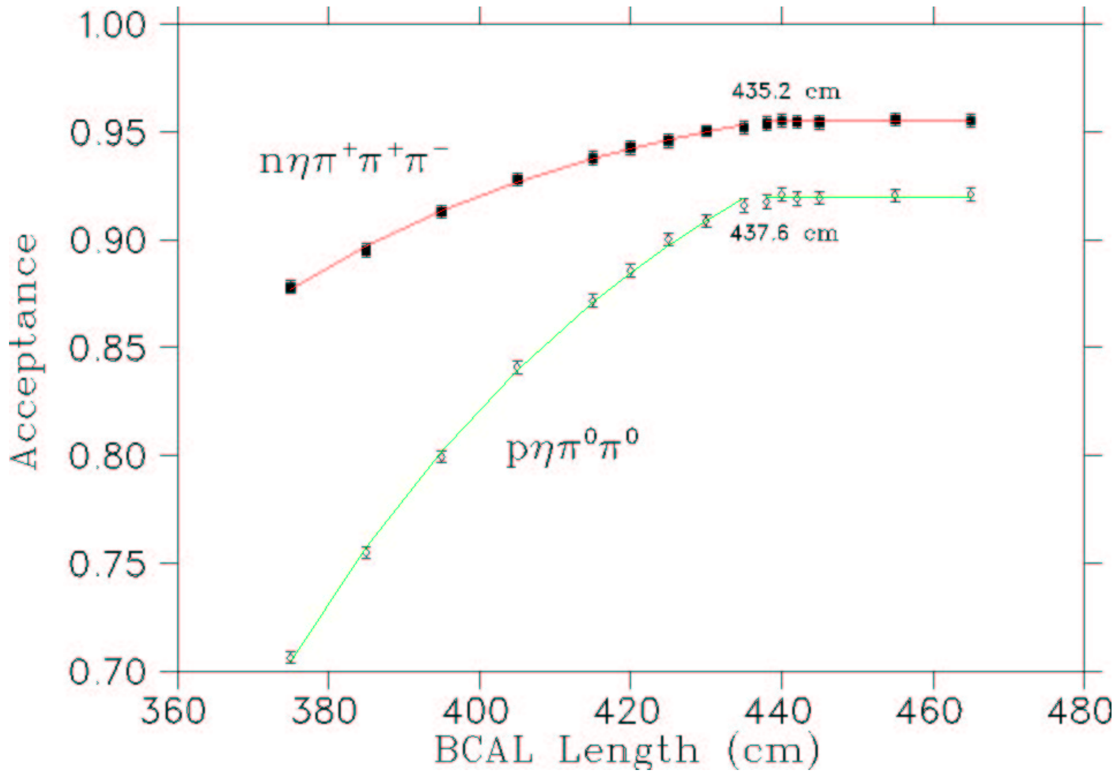

FIG. 3: *Acceptance of the detector as a function of BCAL length for two decay channels:* $n\eta\pi^+\pi^+\pi^-$ *and* $p\eta\pi^0\pi^0$.

and v0.z are vertex positions of the particle and p0.x, p0.y and p0.z are the momenta, a projection could be created of the location at which particle zero, a photon, from each event would hit on a plane defined to be 597.5 cm downstream, the location of the LGD. If there were no LGD downstream, holes in the acceptance would appear simply as a circle. However, since the LGD used by HDFast was a square shape, it was expected that an empty circle with a square area of acceptance within would be observed. Figure 4 illustrates exactly what was expected for a length less than the optimized 440 cm.

F. Future Endeavours

While SPARRO was concerned with using HDFast to study the length of the BCAL, there are plans to study many other aspects of the detector. Some of these areas of interest include:

- How position resolution affects momentum resolution

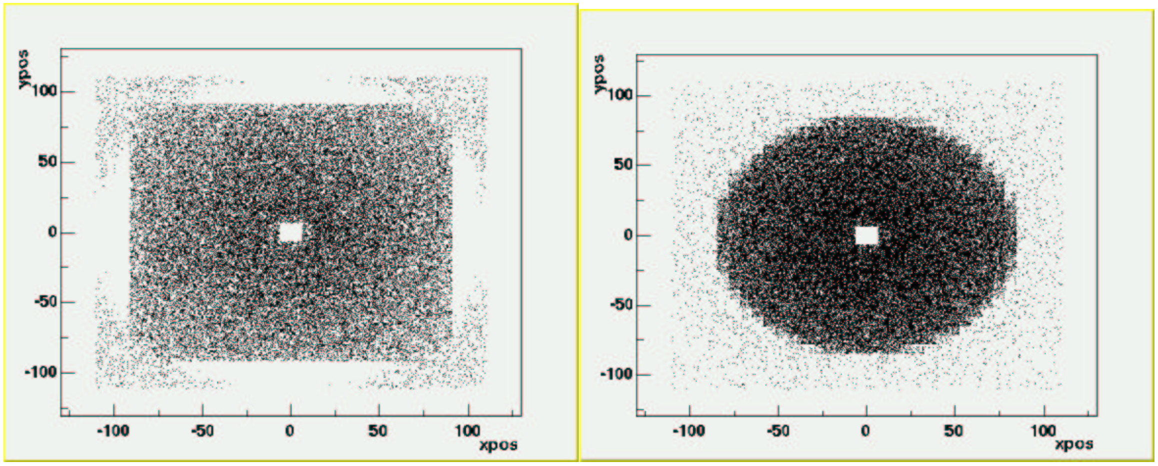

FIG. 4: *Projections of neutral particle hits on an x-y plane for lengths of a) 350 cm b) 465 cm. The former indicates gaps in the acceptance whereas the latter shows complete acceptance.*

- Energy resolution of detector subsystems and overall energy resolution
- Minimum photon energy detected
- Segmentation of detectors
- Vertex resolution for charged particles
- K/π separation
- The affects of resolutions and leakage on PWA

III. GLUEX PORTAL

A. Introduction to the Portal

To allow better communication between all the members of the *GlueX* collaboration, the *GlueX* Portal was designed, downloaded, installed and expanded. A portal is essentially a web site with many features. Search engines, online forums, and internal email are some very common features of a portal. These features make it very advantageous for a large group, like a collaboration, to transfer information regarding all aspects of the experimental progress.

The development of this Portal will be done in three phases. The first phase encompasses the design, graphics, security and the population of the site. During this phase, modules necessary for the desired functionalities were installed. In this phase documentations such as reports, talks, and photos will be available. The second phase will involve porting over of physics simulation modules from another server at the Physics Department in Regina. These modules have been written in PHP, ColdFusion, Perl, Javascript, and HTML. Blocks will be built to specifically carry out these tasks. The third phase will encompass the gradual replacement of procedural modules with object-oriented code and most importantly Grid Services.

The creation of the portal was done using PHP-Nuke 7.0, one of the latest versions of PHP-Nuke [5]. PHP-Nuke is a Content Management System (CMS). CMS is software that permits one to add and/or change content on a web site. PHP-Nuke is written in PHP, a server-side scripting language with syntax based on C and Perl. PHP code can be written within HTML where it can be executed each time a page is visited. PHP-Nuke stores all of its information in a MySQL database [6]. A database allows one to efficiently store, search, sort, and retrieve data. The phpMyAdmin Tool was also utilized [7]. It is a visual system for the management of MySQL databases. It has many useful aspects in that it can be used to create new databases, modify existing databases, perform a dump,and change the contents of a single table or record in the database.

There are many reasons why one would want to use dynamic PHP-Nuke as opposed to static HTML. It allows information to be easily categorized. Within a few pages, a lot of information can be recalled. The dynamic pages allow people to interact via forums, email, and chat. PHP-Nuke simplifies the modification of its standard portal so that one can easily customize it as they wish.

B. Pre-installed Blocks

PHP-Nuke is comprised of blocks, each containing one or more modules. Blocks are the menu items displayed on the right and left panels of the webpages. There are many pre-installed blocks and others are available as add-ons. Some of the most important blocks include the Modules, Administration, and Search blocks.

The Modules block lists all of the active modules as links to the registered user and all modules to the administrator. In other words, as an administrator one can also see those blocks that are inactive or hidden.

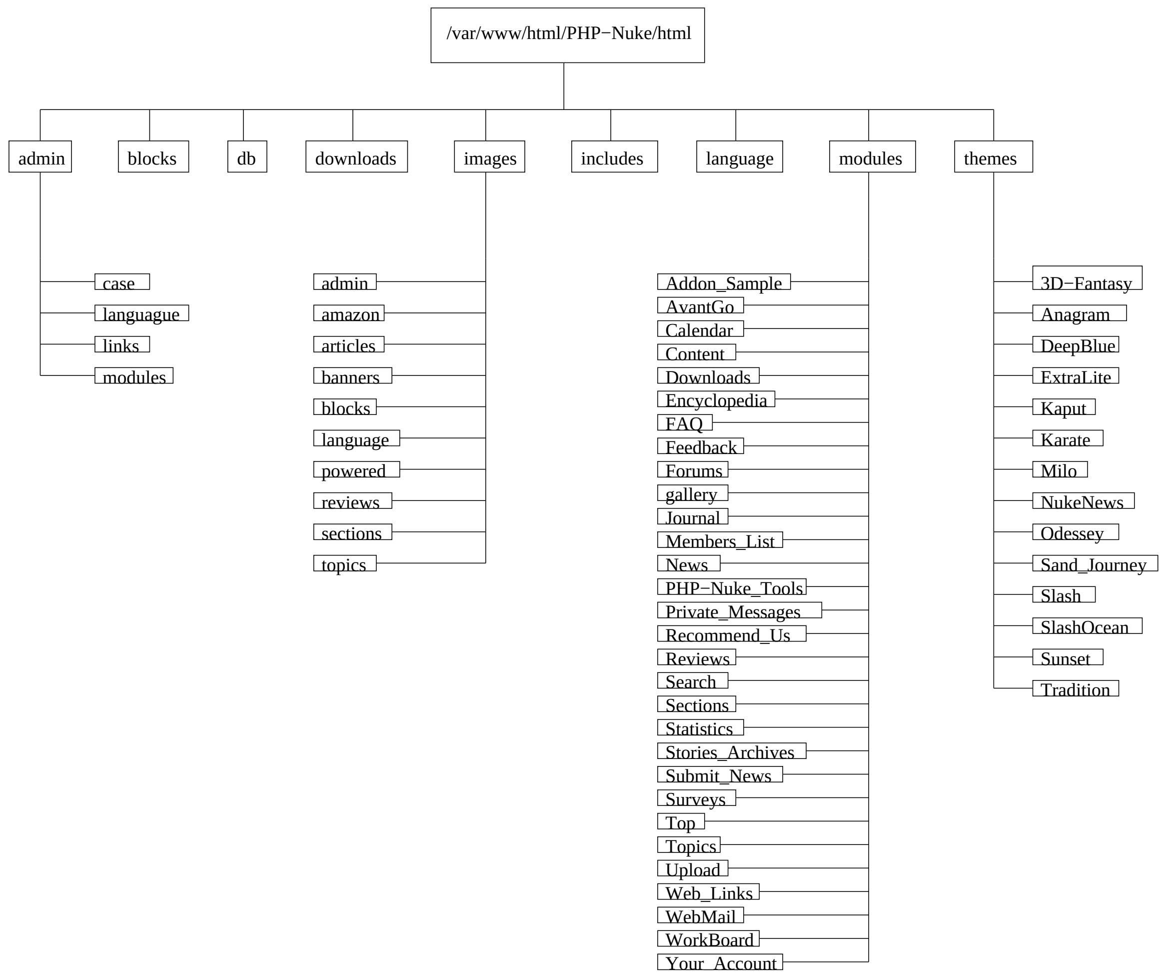

FIG. 5: *Directory structure of the GlueX Portal.*

There are two blocks reserved for use only by the administrator of the site. There is an Administration block which contains links that lead to the administration area. There is also a Waiting Content block. This block displays submissions, reviews, and links waiting for administrative approval. It also shows modules that are not working properly such as broken downloads.

The Search block allows users to search the contents of the website. It makes accessing information quick and easy.

C. Pre-installed Modules

Modules are components of the portal that carry out specific functions. PHP-Nuke comes with many pre-installed modules but has many add-on modules available to allow for individual customization.

The modules are generally easy to install. They require one to install code and then run database queries either from phpMyAdmin or from the command line. As a precaution, the database was saved before and after a new module had been added. To save a database using phpMyAdmin one has to "Export" it and save it in the desired format (e.g. as an sql script). One can also save the database from the Portal's Administration Menu by clicking on the "Backup Database" icon and following the instructions presented.

There are 22 pre-installed modules each with its own purpose. For the *GlueX* Portal some modules are more relevant than others. The administrator of the website has the ability to activate or deactivate modules. PHP-Nuke offers role-based authentication: the administrator can render modules available to all users, registered users or administrators only. There are 12 primary pre-installed modules. These modules incorporate publication, communication and other useful functions.

1. Publication Modules

The modules relating to publication include News, Topics, Sections, Stories Archive, Content, and Submit News.

- The **News** module displays a collection of the most recent articles on the home page of the portal. Articles on the portal are classified according to their topic and category and do not follow a tree-like structure. Categories acts as a cross-reference to topics; they are not subcategories of topics.

- The **Topics** module lists the main classification of material on the portal. This module contains a search interface.

- **Sections** is a classification system parallel to Topics. It can be used for larger articles, and in our case is used as a repository of information that is not looked up frequently.

- The **Stories Archive** module lists articles chronologically. It organizes the articles by the month and year in which the article was submitted.

- The **Content** module is used for full-length stories. It allows the administrator to customize the appearance of articles on the page.

- A user can easily submit an article using the **Submit News** module. The submitter can decide on the article's title, topic, language and text. However, before the article is published, the administrator must approve the article. Administrators are able to display the article on the homepage if they so choose.

2. Communication Modules

Member List, Private Messages, and Forums are modules that can be applied to the communication between registered users.

- The **Member List** module displays all of the registered users and their primary information.

- **Private Messages** is an internal messaging system. Users can save and delete messages they receive.

- The **Forums** module allows for members or groups to have an online forum with threaded messages. This module is very convenient for members of the collaboration.

- The **Your Account** module is indirectly related to communication. The administrator of the site can broadcast messages for all registered users to see from this module.

3. Other Relevant Modules

Downloads and Weblinks are the remaining essential modules for the *GlueX* Portal.

- The **Download** module allows users to add documents for downloading. It contains an inner search function creating easy access to information wanted. This module will be primarily used for material that is large in size. For instance, this will include Hall D Notes, hardware reports, a talk archive and Autocad drawings.

- The **Weblinks** module contains a collection of links. These links relate to the *GlueX* Project, collaborating institutions' web sites and other similar physics research.

D. Add-On Modules

Along with the pre-existing modules, five additional modules were added to the *GlueX* Portal: Approve Membership, WorkBoard, Calendar, Upload, PHP-Nuke Tools, and Gallery. The installation of these modules completed the first phase of the portal.

1. *Approve Membership*

The Approve Membership module allows the administrator of the portal to approve membership applications. After a user fills out an application, their details are sent to the *pending_users* table in the database upon which the administrator receives a notification email. The administration makes the decision whether or not the user should be approved. If a user is rejected, the administration has the option of customizing a rejection email. If a user is approved, the administration can send a customized email to that applicant as well. Within this email, the administration may or may not send an activation link. Upon approval, the user is then stored in the *temp_users* table in the database. When a user receives an approved email with an activation link, they can go to this link and login with their nickname and password. At this point, they become registered users and their information is moved from the *temp_users* table to the *users* table in the database. More than one user can be approved or rejected at one time. Aside from the user's registration form, the administrator also has the option of pre-approving known email addresses. This module easily manages the registration of users which is great for portals with many users. It also contributes to the overall security of the site.

2. *WorkBoard*

The WorkBoard module is operated by the administrator. Its function is to organize work-related tasks. For instance, the administrator can keep track of all workers and assign various roles to them. The administrator can also set up projects and indicate who is involved within a specific project. Tasks that need to be accomplished can also be categorized by timeline, status or priority.

3. Calendar

The Calendar module keeps track of upcoming events. There are five different calendar options. The first shows events for a specific day. The second is a monthly calendar which lists daily events. The third calendar allows one to scroll through upcoming events. The fourth is a monthly calendar with links and a day/month view selection. The fifth calendar is an enhanced version of the scrolling calendar where it only shows the next ten events. The option one uses depends on personal preference. To submit an event, a user must be registered. This is done to prevent a cluttered calendar filled with events unrelated to the collaboration. The administrator approves all calendar entries. This module can greatly help with deadlines and collaboration meetings.

4. Upload

The functionality of the Upload module is fairly clear. It allows registered users to upload files to virtual folders from which users may later download. The Upload module allows the same file name to be used several times in one folder. Each file and directory is owned by one registered user. Only the administrator has the option of creating new folders and assigning them to a registered user. The administrator can also change the ownership of exiting files and folders. When uploading to a folder, the uploader can choose the initial permissions of the file. The permission options are: no permission, read, and read/write. There is an available flag which will grant direct download access for anonymous users. On the *GlueX* Portal, Upload will be used primarily to upload code or restricted-access documents to the server. This code can then be complied, linked and executed with the tools from the second phase.

5. PHP-Nuke Tools

PHP-Nuke Tools module was installed so that new modules and blocks could be created easily thereby allowing increased functionality if needed. Within the module, there are help files for both the module and block creators. PHP-Nuke Tools has many other features. It performs conversions between HTML and other standard web languages including PHP, ASP, JSP, and Perl. One can also create scroll bars, popups, and Meta Tags using PHP-

Nuke Tools. This module may prove to very advantageous in the future to developers and administrators.

6. Gallery

The Gallery module is accessible to all users, registered and anonymous. The *GlueX* Portal will use Gallery to store various images and pictures related to the *GlueX* project such as digital photos of the progress and 3-Dimensional drawings of Hall D. By using the Gallery module, the administrator can create albums and nested sub-albums. These albums can be sorted based on title, popularity and photo capture date. The administrator can grant permissions on these albums so they are only accessible to certain individuals. The administrator can add captions to the images as well as edit multiple captions at once. Photos can be rotated and the thumbnail image can be edited.

IV. CONCLUSION

Many tasks were accomplished this past winter for the SPARRO Group. The HDFast software had some major adjustments to be made but most of these were achieved. Through the use of simulations and x-y plane projections, an optimum BCAL length of 440 cm was found. In the future, there are a few more areas that should be further tested in regards to the BCAL as outlined in the report. Furthermore, hdds-mcfast still has bugs that need to be worked out. For the *GlueX* Portal, the majority of the first phase was completed allowing the remaining phases to receive more focus in the future. The desired use of each module was determined. The population of the site also progressed greatly. At this rate, Phase-1 of the portal should be complete by the end of June 2004.

V. ACKNOWLEDGEMENTS

Blair wishes to thank Dr. Papandreou, Dr. Brash and especially Dr. Huber for the time and effort put into the HDFast Simulations. Rheanne would also like to thank Dr. Papandreou for his supreme knowledge and guidance with the portal.

VI. BIBLIOGRAPHY

[1] http://cepa.fnal.gov/psm/simulation/mcfast/

[2] *ROOT Tutorials from CERN*

http://root.cern.ch/root/Tutorials.html

[3] *R.T. Jones, University of Connecticut, private communication (May 2004).*

[4] *P. Eugenio, A Study of Acceptance for the Stage 1 Hall D Detector, Note 16 (1999).*

http://www.phys.cmu.edu/halld/halld_notes/index_notes.html

[5] http://www.phpnuke.org

[6] http://www.mysql.com/

[7] http://www.phpMyAdmin.net

VII. APPENDIX A: SIMULATE SCRIPT

```
#!/bin/sh
# parameters reaction, evno, fname are passed in the command line:
# simulate rhop 10000 test
# Authors: ZP and EB
#
if [ -z "$1" ]; then
  echo 'usage: simulate reaction [noofevents] [filename] [xmass] [xwidth] [beamenergy]'
  echo 'defaults: Xmass=1.7 Xwidth=0.3 Beamenergy=8.0 noofevents=1000 resultsfilename=test'
else
  reaction=$1
  if [ -z "$4" ]; then
    xmass=1.7
  else
    xmass=$4
  fi
  if [ -z "$5" ]; then
    xwidth=0.3
  else
    xwidth=$5
  fi
  if [ -z "$6" ]; then
    beamenergy=8.0
  else
    beamenergy=$6
  fi
  if [ -z "$2" ]; then
    evno=1000
  else
    evno=$2
  fi
  if [ -z "$3" ]; then
    startfname=test
  else
    startfname=$3
  fi
  case $reaction in
n_3pi ) particles=3 ;;
omegadelta2 ) particles=6 ;;
rhop ) particles=3 ;;
pk+k-pi+pi- ) particles=5 ;;
p_K-pi+pi-K+ ) particles=5 ;;
p_pi+pi-pi0 ) particles=5 ;;
n_pi-pi+pi+ ) particles=4 ;;
n_eta_pi+pi-pi+ ) particles=6 ;;
n_omega_pi0_pi+ ) particles=8 ;;
n_omega_pi+ ) particles=6 ;;
p_eta_pi0pi0 ) particles=7 ;;
* ) echo "reaction: $reaction is an unknown reaction type."
    exit 1 ;;
  esac
#  fname=$startfname'_R'$reaction'_M'$xmass'_W'$xwidth'_E'$beamenergy

fname=$reaction'_B'$beamenergy'_M'$xmass'_W'$xwidth'_'$startfname
#   fname=$startfname'_'$reaction
  cp $reaction.input temp.input
  echo 's/beamZ/'$beamenergy'/g' > sedscr
  echo 's/Xmass/'$xmass'/g' >> sedscr
  echo 's/Xwidth/'$xwidth'/g' >> sedscr
  runsed temp.input
  cp generic.cmd $fname.cmd
  echo 's/xyz/'$fname'/g' > sedscr
  runsed $fname.cmd
  echo 'Environment Variables .....'
  which HDFast
  which genr8
  which ntp_maker

  genr8 -M$evno -A$fname.ascii < temp.input
  rm -f temp.input
  ascii2stdhep -i$fname.ascii -o$fname.evt -N$evno -n$particles

  HDFast -f $fname.cmd -o $fname.rdt
  #run_hdfast -N$evno -t$fname -n$particles
  ntp_maker -c $fname.rdt
  mv halld.root $fname.root
  rm -f *.bak
fi
```

VIII. APPENDIX B: SUPERDB.DB

```
database mcfast 0000
!
! geo_cdc_v02.db   Hall D Meson Spectrometer
!                  Straw Tube CDC Design
!
! Created By:      Paul M. Eugenio & Curtis A. Meyer
!                  Carnegie Mellon University
! Updated:         13 June, 2000
!                  (Made consistent with Eric Scott's Drawings)
!
!                  The coordinate system has z along the photon beam, y
!                  is defined as vertically up, and x makes the system
!                  right handed.
!
!                  The origin is on the beam axis at the center of the
!                  up stream mirror plate.
!
!      Details of the geometry description for MCFast can be
!                  found at:
!      http://www.phys.cmu.edu/halld/track/doc/doc_index.html
!
!  notes:          Comment lines start with a "!".
!    Lines starting with a "!%" are special comments
!    which act as tags for the geomerty viewer.
!
! ***********  Define the detector
!
include db/detector.db
make detector "GLUEX" "CENTRAL"
!
! ************ Define New Materials.
!
include db/materials.db
! ***  CF4 = Freon for Chamber gas
make MATERIAL "CF4"  88.  42.  3.9286E-3 64000. 5.0E+04 1.0E+16 6.75E-06
!
! ************ Define New Mixtures
include db/mixtures.db
! *** Chamber Gas
make MIXTURE  "AR-ETH-CF4" 3 "ARGON"  "C2H6"  "CF4"  "-"  "-" 0.5 0.35 0.15
! *** Chamber Gas with including Al. wire.
make MIXTURE  "GAS-MX" 3 "ARGON"  "C2H6"  "CF4"  "-" "-" 0.500 0.35 0.15
! *** Lead - Scintillator Sandwich for Calorimeter
make MIXTURE  "PBSC"       2 "LEAD"  "SCIN"  "-"  "-"  "-" 0.4 0.6

!
! ************** Create the Beam pipe
!
include db/beampipe.db
!
! ***.05 cm thick Be pipe at a radius of 2.50cm which runs from z=0 to z=450.
!
!          name  rmin rmax   z0  zlen   fill  rmin rmax front back  fill  wall  front   back
make BPipe "beam" 0.00 2.50 275.0 550.0 "Vacuum" 0.00 0.05 0.00  0.00 "Vacuum" "Beryllium"  "Vacuum" "Vacuum"
!
! *************** The MEGA-LASS Magnetic Field
! *** 2.24 T field
! *** 420cm long field volume. Bore is 415cm long.
! *** 95cm.
!
include db/solenoid.db
!
!              name  bfield  rmin      rmax       z0    zlen    FILL  rmin   rmax   zmin    zmax     fill     wall    front   back
make Solenoid  "LASS" 2.24000 0.0000   95.0000    196.6  494.8   "Air" 0.0000 2.0000 0.0000  0.0000  "Vacuum"  "Iron"  "Iron"  "Iron"
!
!
! *********************** Barrel Calorimeter
!
!  Pb-SciFi Barrel Calorimeter starts at z=15cm, and extends to z=420cm.
!  It extends from r=65cm to r=90cm.
!
include db/emcal.db
make EMCal "BCAL" "TUBE" 1 65. 65. 90. 90. 235.0 440.0 "leadScint" "leadScint" 48 1 0.06 0.01 0.3 0.03 4.0 1
!
!
! ************************ LGD
!
! LGD is hard coded into HDFast. See usr_lgd.c
! But the "!%" is used by GeoDraw.C to draw the LGD
!include db/calorbox.db
!% make  CalorBox "LGD" "BOX" 2 -95 95 -95 95 0 0 0 0 597.5 45 "PBGL"  "PBGL" 71 43 0.06 0.01 0.0 0.0 4 1
!
!
! *********************** VTX Vertex Chamber Outline
!
include db/drift.db
!
!  This is a kludged version of the start counter.
!  It starts at z=40cm and extends to z=130 cm.
!  the target exyends from z=50cm to z=80cm
!
!          #  name  ly cth rmin  rmax   z0      zl     fill   rmin   rmax  front   back   materials
make Drift 1 "VTX"  3  0   4.95  5.65   80.0  90.000  "Scintillator" 0.2000 0.0500 0.2000 0.2000  "Mylar" "Mylar" "Mylar" "Mylar"
! *************************** Central Drift Chamber
!
!
! *** CDC Chamber Outline
!
!    the straw tube chamber is 2m long,
!    and starts at z=15 and extends to z=215
!
!          #  name  ly cth rmin  rmax   z0      zl     fill   rmin    rmax   front   back     materials
make Drift 2 "CDC" 22  0  15.0  60.0   117.0  200.0  "Scintillator" 0.2000 0.50 1.10  1.10  "CarbonFiber" "CarbonFiber" "Aluminum" "Aluminum"
!
!
!
include db/drift_layer.db
!
!          three layers for the start counter:
!
!          drift#  anode  rad   zlen  c_sz #wr  IDr IDc phi0 st_t st_of eff_hit eff_dedx siga sib sigc
make LayerDRFAno 1   1     5.0   90.0  1.0   25  -1  1   0.  -0.10   0.  .96     0.96     .04  0.0 0.0
make LayerDRFAno 1   2     5.1   90.0  1.0   25  -1  1   0.   0.0    0.  .96     0.96     .04  0.0 0.0
make LayerDRFAno 1   3     5.2   90.0  1.0   25  -1  1   0.   0.10   0.  .96     0.96     .04  0.0 0.0
!
!
!
! make anode layers
!
!          22 1cm radius tubes for the CDC
```

```
!           drift#  anode   rad    zlen   c_sz #wr   IDr IDc phi0 st_t st_of eff_hit eff_dedx siga sib sigc
make LayerDRFAno 2 1 16.049 200.0 1.6 63 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 2 17.831 200.0 1.6 70 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 3 19.613 200.0 1.6 77 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 4 21.395 200.0 1.6 84 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 5 23.178 200.0 1.6 91 -1 0 0. 0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 6 24.960 200.0 1.6 98 -1 0 0. 0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 7 26.742 200.0 1.6 105 -1 0 0. -0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 8 28.524 200.0 1.6 112 -1 0 0. -.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 9 32.089 200.0 1.6 126 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 10 33.871 200.0 1.6 133 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 11 35.654 200.0 1.6 140 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 12 37.436 200.0 1.6 147 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 13 39.218 200.0 1.6 154 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 14 41.001 200.0 1.6 161 -1 0 0. 0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 15 42.783 200.0 1.6 168 -1 0 0. 0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 16 44.566 200.0 1.6 175 -1 0 0. -0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 17 46.348 200.0 1.6 182 -1 0 0. -0.105 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 18 49.149 200.0 1.6 193 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 19 50.932 200.0 1.6 200 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 20 52.714 200.0 1.6 207 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 21 54.497 200.0 1.6 214 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 22 56.279 200.0 1.6 221 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00
make LayerDRFAno 2 23 58.062 200.0 1.6 228 -1 0 0. 0.0 0. 0.96 0.96 0.02 0.00 0.00


!
include db/drift_offset.db
!
!  *** Offsets to drift layers -- not used.
!
make OffsetDRFAno  1   1  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  1   2  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  1   3  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
!
! ***********---- END OF VTX ----**********
!
make OffsetDRFAno  2   1  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   2  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   3  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   4  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   5  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   6  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   7  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   8  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2   9  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  10  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  11  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  12  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  13  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  14  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  15  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  16  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  17  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  18  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  19  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  20  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  21  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
make OffsetDRFAno  2  22  0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

!
! ** cdc cathode strips:
!include db/drift_cathode.db
!
!   do not enable cathode strips on the straw tube chamber.
!
!make LayerDRFCatho 2 1 0.1 200. 200 6   1  0. 0.96 0.050 0. 0.
!make LayerDRFCatho 2 2 0.1 200. 200 12 16  0. 0.96 0.050 0. 0.
!
!
!** extra CDC materials
!
! Place some Aluminum around the cylindrical chambers:
! Place some Carbon Fiber inside the chambers
!
! The following absorber material simulates AL foil for straws in the
! CDC chamber
!
!              name    shape  ty  rmin  rmin  rmax  rmax  z0   zlen    FILL
include db/absorber.db
make Absorber "SHIT01" "TUBE" 41 21.00 21.00 21.04 21.04 117.0 200.0 "Aluminum"
make Absorber "SHIT03" "TUBE" 41 25.00 25.00 25.04 25.04 117.0 200.0 "Aluminum"
make Absorber "SHIT05" "TUBE" 41 29.00 29.00 29.04 29.04 117.0 200.0 "Aluminum"
make Absorber "SHIT07" "TUBE" 41 31.00 31.00 31.04 31.04 117.0 200.0 "Aluminum"
make Absorber "SHIT09" "TUBE" 41 35.00 35.00 35.04 35.04 117.0 200.0 "Aluminum"
make Absorber "SHIT11" "TUBE" 41 37.00 37.00 37.04 37.04 117.0 200.0 "Aluminum"
make Absorber "SHIT13" "TUBE" 41 41.00 41.00 41.04 41.04 117.0 200.0 "Aluminum"
make Absorber "SHIT15" "TUBE" 41 45.00 45.00 45.04 45.04 117.0 200.0 "Aluminum"
make Absorber "SHIT17" "TUBE" 41 49.00 49.00 49.04 49.04 117.0 200.0 "Aluminum"
make Absorber "SHIT19" "TUBE" 41 53.00 53.00 53.04 53.04 117.0 200.0 "Aluminum"
make Absorber "SHIT21" "TUBE" 41 57.00 57.00 57.04 57.04 117.0 200.0 "Aluminum"

!
! ***********---- END OF CDC ----**********
! ** extra FDC MATERIALS
!
!  The following puts support material around the forward chambers
!
!              name    shape  ty  rmin  rmin  rmax  rmax  z0   zlen    FILL
make Absorber "SHL7" "TUBE" 41 60.0 60.0 61.0 61.0 230.0 12.0 "Aluminum"
make Absorber "SHL8" "TUBE" 41 60.0 60.0 61.0 61.0 282.0 12.0 "Aluminum"
make Absorber "SHL9" "TUBE" 41 60.0 60.0 61.0 61.0 338.0 12.0 "Aluminum"
make Absorber "SHL10" "TUBE" 41 60.0 60.0 61.0 61.0 394.0 12.0 "Aluminum"

!
! Note that KludgeAnnulus is not a MCFast geometry type.
!                        name      Rmin  Rmax  zLength  zCenter
!% make KludgeAnnulus "CERENKOV"  20.0  95.0  90.0   455.0
!
!
! ********************************* TOF & CERENKOV
!           --> FORWARD TOF & CERENKOV <--
!
!  The forward time-of-flight wall is treated as material centered at
!  z=566.27 cm, and 2.54 cm thick.
!
include db/absorberbox.db
make AbsorberBox "FTOF"  "BOX"  42  -125.0  125.0  -125.0  125.0 -6.0 6.0 -6.0 6.0 566.27 2.54  "Scintillator"

!
!  The cherenkov detector is treated as a "plane" where the charged
!  particle enters the detector. It is placed as plane at 420.5 cm
!  and is 1cm thick.
!
make AbsorberBox "Cerenkov"  "BOX"  2  -95.0  95.0  -95.0  95.0 -6.0 6.0 -6.0 6.0 420.5 1.0  "Scintillator"
```

```
!
! The "!%" is used by GeoDraw.C  and EventDisplay.C to draw the CERENKOV
!
!  ***************************** FORWARD DISK Chambers
!
include db/sidisk.db
!                  (not really Silicon).
!            #  name  lyr  zpos
make SiDisk  1 "FDC1"   9  227.0
make SiDisk  2 "FDC2"   9  233.0
make SiDisk  3 "FDC3"   9  279.0
make SiDisk  4 "FDC4"   9  285.0
make SiDisk  5 "FDC5"   9  335.0
make SiDisk  6 "FDC6"   9  341.0
make SiDisk  7 "FDC7"   9  391.0
make SiDisk  8 "FDC8"   9  397.0
!
include db/sidisk_layer.db
!
! ** make anode layers
!
!             det dsk Material wed  zoff thick  rmin  rmax phi_rng dphi  type
! anodes
make LayerSiDi  1  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  1  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  1  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  1  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  1  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  1  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  1  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  1  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  1  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  2  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  2  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  2  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  2  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  2  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  2  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  2  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  2  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  2  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  3  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  3  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  3  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  3  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  3  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  3  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  3  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  3  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  3  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  4  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  4  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  4  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  4  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  4  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  4  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  4  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  4  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  4  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  5  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  5  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  5  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  5  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  5  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  5  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  5  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  5  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  5  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  6  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  6  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  6  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  6  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  6  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  6  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  6  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  6  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  6  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  7  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  7  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  7  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  7  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  7  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  7  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  7  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  7  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  7  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
make LayerSiDi  8  1  "chamberGas"  1  -2.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  8  2  "chamberGas"  1   0.0  2.00  3.50 60.00 0. 360 360 3
make LayerSiDi  8  3  "chamberGas"  1   2.0  2.00  3.50 60.00 0. 360 360 3
! cathodes
make LayerSiDi  8  4  "Mylar"  1  -2.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  8  5  "Mylar"  1  -1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  8  6  "Mylar"  1  -0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  8  7  "Mylar"  1   0.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  8  8  "Mylar"  1   1.5  0.008 3.50 60.00 0. 360 360 3
make LayerSiDi  8  9  "Mylar"  1   2.5  0.008 3.50 60.00 0. 360 360 3
!
include db/sidisk_wedge.db
!
!
!Wedges for the disk drift chambers
!
! Each package(det) contains 9 2pi-Wedges (3 anode layers(x,u,v)
! + 2 cathode strip layers for each anode layer).
!
!          speci det lyr nwed nstrp c0_r  c0_f pitch stereo eff  resolutions
! anode layers
make Wedge "ALL"  1   1   1   119   0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  1   2   1   119   0.000 0.00 1.0   60.0   0.96  0.015  0. 0.
make Wedge "ALL"  1   3   1   119   0.000 0.00 1.0  -60.0   0.96  0.015  0. 0.
! cathode strips
```

```
make Wedge "ALL"  1   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  1   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  1   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  1   7   1   119    0.000 0.00 1.0   0.00    0.96  0.015  0. 0.
make Wedge "ALL"  1   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  1   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  2   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  2   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  2   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  2   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  2   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  2   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  2   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  2   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  2   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  3   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  3   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  3   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  3   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  3   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  3   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  3   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  3   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  3   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  4   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  4   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  4   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  4   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  4   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  4   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  4   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  4   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  4   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  5   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  5   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  5   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  5   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  5   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  5   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  5   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  5   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  5   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  6   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  6   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  6   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  6   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  6   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  6   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  6   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  6   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  6   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  7   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  7   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  7   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  7   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  7   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  7   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  7   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  7   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  7   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
make Wedge "ALL"  8   1   1   119    0.000 0.00 1.0   0.000    0.96  0.015  0. 0.
make Wedge "ALL"  8   2   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  8   3   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
! cathode strips
make Wedge "ALL"  8   4   1   119    0.000 0.00 1.0   60.0    0.96  0.015  0. 0.
make Wedge "ALL"  8   5   1   119    0.000 0.00 1.0  -60.0    0.96  0.015  0. 0.
make Wedge "ALL"  8   6   1   119    0.000 0.00 1.0   120.0   0.96  0.015  0. 0.
make Wedge "ALL"  8   7   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  8   8   1   119    0.000 0.00 1.0   0.0000   0.96  0.015  0. 0.
make Wedge "ALL"  8   9   1   119    0.000 0.00 1.0  -120.0   0.96  0.015  0. 0.
!
!
! ***********---- END OF FDC ----***********
!
include db/hitsontrack.db
!
!                all  z  svx
make HitsOnTrack  4   0   0
!
! **************************** Target Vertex
!
!
include db/beamvrtx.db
!           x   y    z     sigx     sigy   sigz
make BeamVrtx  0.  0.  65    0.30     0.30    15.
!
end
```