

Detector simulation with GSTAR and
root4star

*Tutorial at the Dubna regional STAR
meeting, November 2003*

Mikhail Kopytine (Kent State University)

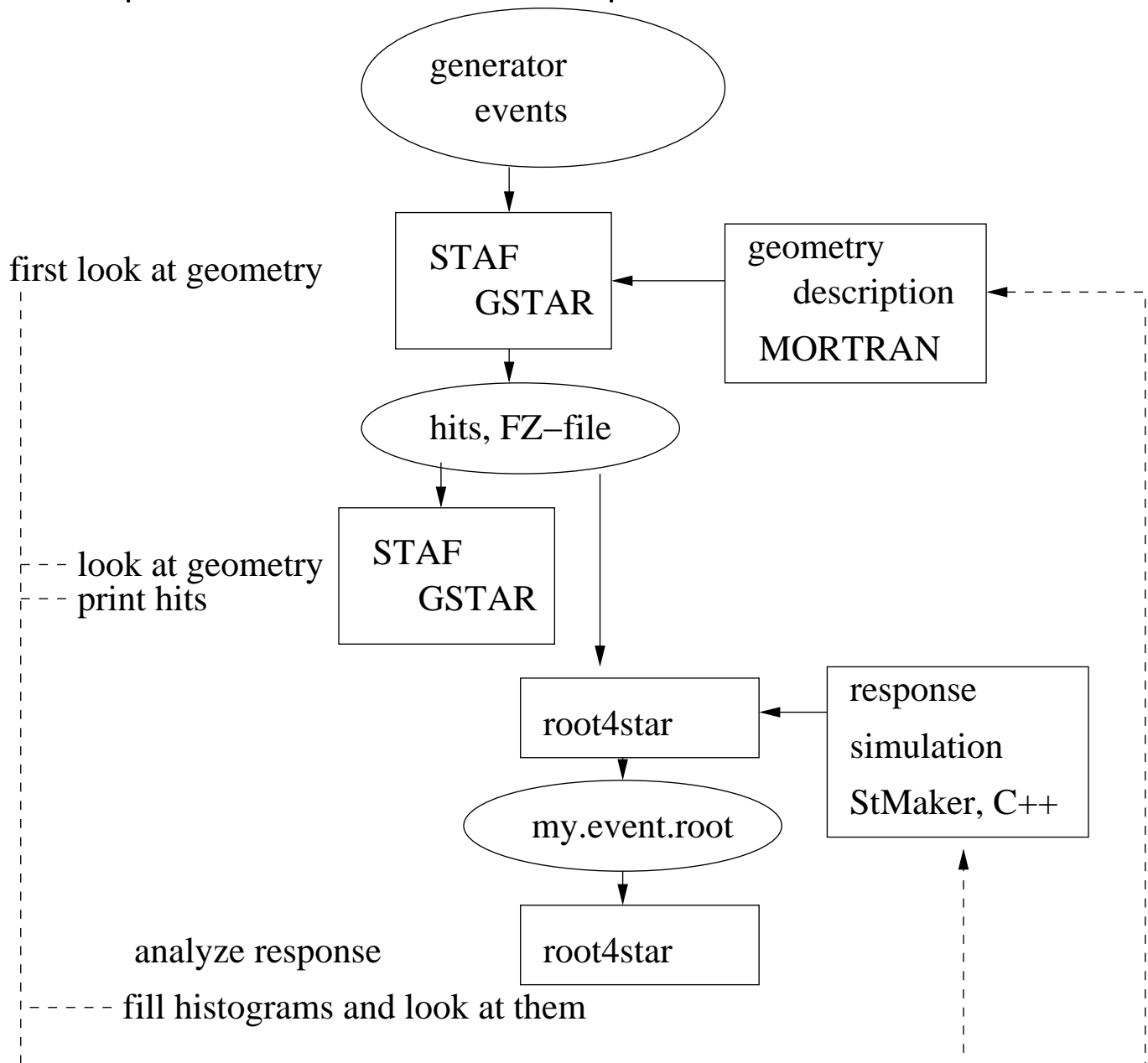


Key design decisions made in STAR MC:

- GEANT 3.21 \Rightarrow age of FORTRAN: CERN-LIB, PAW, ZEBRA (old good products)
 - detector geometry description relies on MORTRAN: a public domain FORTRAN preprocessor for structured programming.
 - use root4star environment (“Big Full Chain” or BFC) for detector response (hit digitization) \Rightarrow independent second stage, based on StMaker (C++).
- \Rightarrow heterogeneous system; “hidden” tools to take care of legacy FORTRAN/C code; things to learn for non-ambidextrous FORTRAN/C++ users.



This talk will show you how to make these steps in the code development:



```

rxvt
/bscmgeo.g
*****
Module BBCMGEO is the Beam Beam Counter Modules GEOMETRY
Created 15 March 2002
Author Yiqun Wang
*
*****
* Beam pipe has a diameter of 4cm
*
*****
+CDE,AGECOM,GCONST,GCUNIT.
*
Content      BBCM,BBCA,THXM,SHXT,BPOL,CLAD
*
Structure    BBCG {version,onoff(3),zdis(2)}
*
Structure    HEXG {type,irad,clad,thick,zoffset,xoffset,yoff
set}
*
Real        actr,snad,lrad,ztotal,x0,y0,theta0,phi0,
xtrip,ytrip,rtrip,thetrip,rsing,thesing
*
Integer      I_trip,J_sing
*
-----
*
Fill      BBCG          ! BBC geometry
          Version = 1.0 ! Geometry version
          Onoff = {3,3,3} ! 0 off, 1 west on, 2 e
ast on, 3 both on: for BBC,Small tiles,Large tiles
          zdis = {374,24,-374,24} ! z-coord from center i
n STAR (715/2+6*2.54+1=373.8)
*
-----
*
Fill      HEXG          ! hexagon tile geometry
          Type = 1      ! 1 for small hex tile,
          irad = 4.174  ! inscribing circle rad
          clad = 0.1    ! cladding thickness
          thick = 1.0   ! thickness of tile
          zoffset = 1.5 ! z-offset from center of BBCU
          xoffset = 0.0 ! x-offset center from beam for
          BBCU (1), or BBCE (2)
          yoffset = 0.0 ! y-offset center from beam for
          BBCU (1), or BBCE (2)
*
Fill      HEXG          ! hexagon tile geometry
          Type = 2      ! 1 for small hex tile,
          irad = 16.697 ! inscribing circle rad
          clad = 0.1    ! cladding of tile
          thick = 1.0   ! thickness of tile
          zoffset = -1.5 ! z-offset from center of BBCU
          xoffset = 0.0 ! x-offset center from beam for

```

```

rxvt
C: definitions from /afs/rhic/star/packages/StAF/SL00a/.i386_redhat61/bin/geant
t3_def
*****
SUBROUTINE BBCMGEO 2
C BBCMGEO IS THE BEAM BEAM COUNTER MODULES GEOMETRY 2
IMPLICIT NONE 2
CHARACTER MODULE*20,MCOMMENT*80,BANK_TITLE*60 2
DATA MODULE/'BBCMGEO'/,BANK_TITLE/'/'/ 2
DATA MCOMMENT/'IS THE BEAM BEAM COUNTER MODULES GEOMETRY'/ 2
LOGICAL FIRST 2
DATA FIRST/.TRUE./ 2
SAVE FIRST 2
INTEGER JUMPAD 2
EXTERNAL JUMPAD 2
INTEGER OK 2
PARAMETER (OK=0) 2
INTEGER AG_I,AG_J 2
SAVE AG_I,AG_J 2
C CREATED 15 MARCH 2002 2
DATA AG_I/0/,AG_J/0/ 3
CHARACTER*40 CREATED /'15 MARCH 2002'/ 3
C * . . . . . * 3
C AUTHOR YIQUN WANG 3
CHARACTER*40 AUTHOR /'YIQUN WANG'/ 4
C * . . . . . * 4
*
* Beam pipe has a diameter of 4cm
*****
C - Advanced Geant interface 11
CHARACTER*20 AG_MODULE,AG_TITLE,AG_EXNAME,AG_PARLIST,AG_CHDIR, 11
*AG_MATERIAL,AG_MIXTURE,AG_COMPONENT,AG_MEDIUM 11
CHARACTER*4 AG_VOLUME,AG_MOTHER,AG_SHAPE,AG_CNICK,AG_KONLY, 11
*AG_OPTION,AG_ORI,AG_MARK 11
INTEGER AG_BEGCOM,AG_IVOLUME,AG_IMOTHER,AG_IGNUM,AG_ISHAPE, 11
*AG_IMED,AG_IMAT,AG_IFIELD,AG_NLMAT,AG_IERROR,AG_NUBUF,AG_NPAR, 11
*AG_ISTATUS,AG_IROT,AG_NBITS,AG_SERIAL,AG_ATTRIBUTE(6),AG_WORK, 11
*AG_SEEN,AG_LSTY,AG_LUID,AG_COLO,AG_FILL,AG_JDU,AG_ORTI,AG_NDIV, 11
*AG_IAXIS,AG_NOVMAX,AG_IPRIN,AG_RESET1,AG_RESET2,AG_BEGSCR, 11
*AG_ENDSCR,AG_IDTYPE,AG_ISET,AG_IDET,AG_ISVOL,AG_LEVEL,AG_IRESER, 11
*AG_IGEOM,AG_STANDALONE,AG_LSTACK,AG_NUOHIT,AG_NUUVOL,AG_MAGIC, 11
*AG_LDETU,AG_NPDIV,AG_NPDV,AG_NZ,AG_NCOPY,AG_IDEBU,AG_IGRAP, 11
*AG_IHIST,AG_IMFLD,AG_ISIMU,AG_ENDSAVE,AG_CODE,AG_TRKTYP,AG_MODE, 11
*AG_ECODE,AG_PDG,IPRIN 11
REAL AG_FIELDM,AG_TMAXFD,AG_STEMAX,AG_DEEMAX,AG_EPSIL,AG_STMIN, 11
*AG_DENS,AG_RADL,AG_ABSL,AG_THETRX,AG_THETRY,AG_THETAZ,AG_ALFAX, 11
*AG_ALFAY,AG_ALFAZ,AG_PHIX,AG_PHIY,AG_PHIZ,AG_ALPHAX,AG_ALPHAY, 11
*AG_ALPHAZ,AG_TYPE,AG_DX,AG_DY,AG_DZ,AG_DY1,AG_DY2, 11
*AG_DMAXMS,AG_TWIST,AG_THET,AG_THET1,AG_THET2,AG_PHI,AG_PHI1, 11
*AG_PHI2,AG_ALPH,AG_ALP1,AG_ALP2,AG_RMIN,AG_RMAX,AG_RMN, 11
*AG_RMX,AG_ZI,AG_RMN1,AG_RMN2,AG_RMX1,AG_RMX2,AG_HI,AG_H2, 11
*AG_BL1,AG_BL2,AG_TL1,AG_TL2,AG_DPFI,AG_DZ,AG_TWIS,AG_X, 11
*AG_Y,AG_Z,AG_A,AG_ZA,AG_U,AG_STEP,AG_CO,AG_PAR,AG_AA, 11
*AG_ZZ,AG_UU,AG_LX,AG_LY,AG_LZ,AG_HX,AG_HY,AG_HZ,AG_P1, 11
*AG_P2,AG_STACK,AG_UBUF,AG_XHMAX,AG_YHMAX,AG_ZHMAX,AG_RHMAX, 11
*AG_FHMAX,AG_FHMIN,AG_BIN,AG_CHARGE,AG_MASS,AG_TLIFE,AG_BRATIO 11
PARAMETER (AG_LSTACK=130,AG_NUOHIT=10,AG_NUUVOL=3,AG_MAGIC= 11
*-696969,AG_LDETU=250) 11
COMMON/AGCLOB/ AG_MODULE,AG_CHDIR,AG_LEVEL,AG_IDTYPE,AG_IERROR, 11
*AG_STANDALONE,IPRIN,AG_IPRIN,AG_IGEOM,AG_IDEBU,AG_IGRAP,AG_IHIST, 11
*AG_IMFLD,AG_ISIMU 11
C Inherited variables a: saved during internal calls 11
C b: reset at the entry in a block and then saved 11
COMMON/AGCPARA/ AG_BEGCOM,AG_IVOLUME,AG_IMOTHER,AG_IGNUM, 11
*AG_ISHAPE,AG_IMED,AG_IMAT,AG_IFIELD,AG_FIELDM,AG_TMAXFD, 11
*AG_STEMAX,AG_DEEMAX,AG_EPSIL,AG_STMIN,AG_DENS,AG_RADL, 11

```

Detector description with Advanced Geant User Interface (left) and a machine-generated one in FORTRAN (right).

The image shows a window titled "plot_geom.kumac" with a menu bar (File, Edit, Options, Buffers, Tools, Help) and a toolbar with icons for file operations. The main area contains a script for drawing a STAR detector geometry. The script includes comments, a title, and various drawing commands like "detp", "make", "dopt", "option", "satt", "box", "next", "gr/del", "dopen", "draw", "dclose", "dshow", "dman", "dhead", "close", and "fort/file".

```
*****
* Example of kumac to draw STAR detector
*****
detp  geometry year_2a

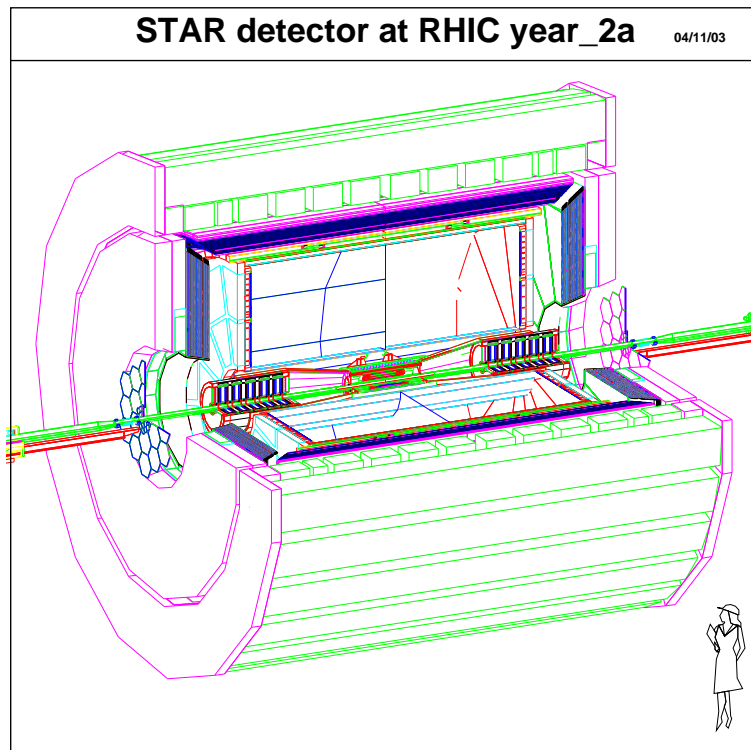
make geometry
  call bbcgeo
gclose all

dopt hide on; dopt shad off ; dopt edge on
option rbox;      option nsta
satt * fill 6;    satt * lwid 1

satt HALL seen 0
satt CAVE seen 0
satt * fill 6
satt * lwid 1
box .
box * -3000. 0. 0. 3000. -2000. 1200.
next
gr/del 1
dopen 1
draw CAVE 115 160 0 10 10 .019 .019
dclose 1
next
dshow 1
dman 17 2.5 wm1
dhead 1101 'STAR detector at RHIC' 0.65
█
*****
*Saving into a file
*****
fort/file 66 starview.ps; meta 66 -111
next; dshow 1; dman 19. 2.5 wm2
dhead 1101 'STAR detector at RHIC year_2a' 0.65
close 66; meta 0
-- plot_geom.kumac (Fundamental) --L29--CO--All-----
```

KUIP: Same syntax for command line typing and scripting.

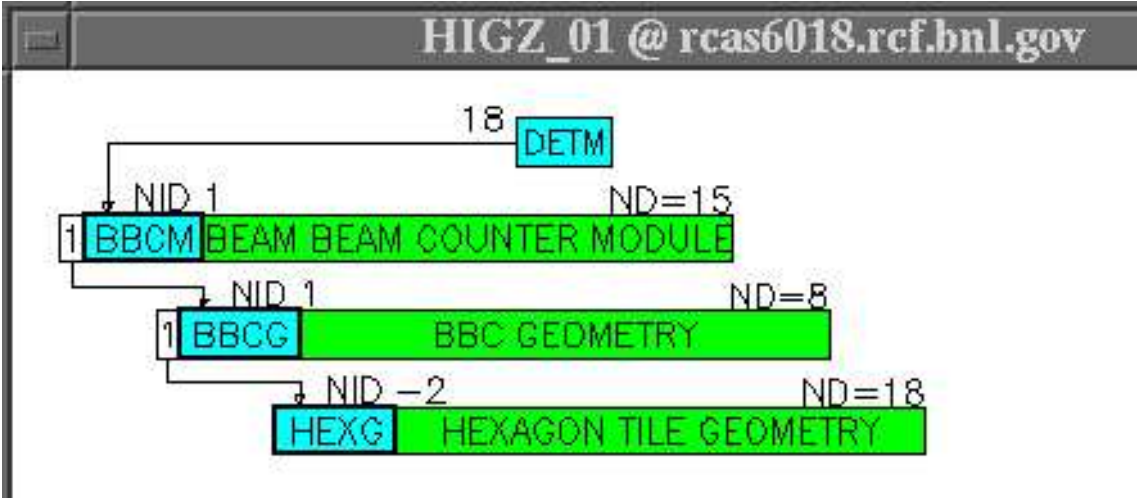
```
rxvt
[rcas6018] /<4>kopytin/trigger/trigger/dAu/> staf -w 1
*****
* Starting      staf NwGEANT= 20000000 NwPAW= 2000000 *
*****pid= 4549*****
*****
*
*           W E L C O M E   to   staf
*
*           Version 2.00/04   7 October 2003
*
*****
Version 1.29/03 of HIGZ started
1***** GEANT Version 3.21/14 Released on 19032002
0***** Correction Cradle Version 0.1400
***** Initialising Staf libraries
***** RZMAKE. OLD RZ format selected for RZDOC
staf > exec plot_geom.kumac
YEAR_2A: OLD ASYMPTOTIC STAR
```

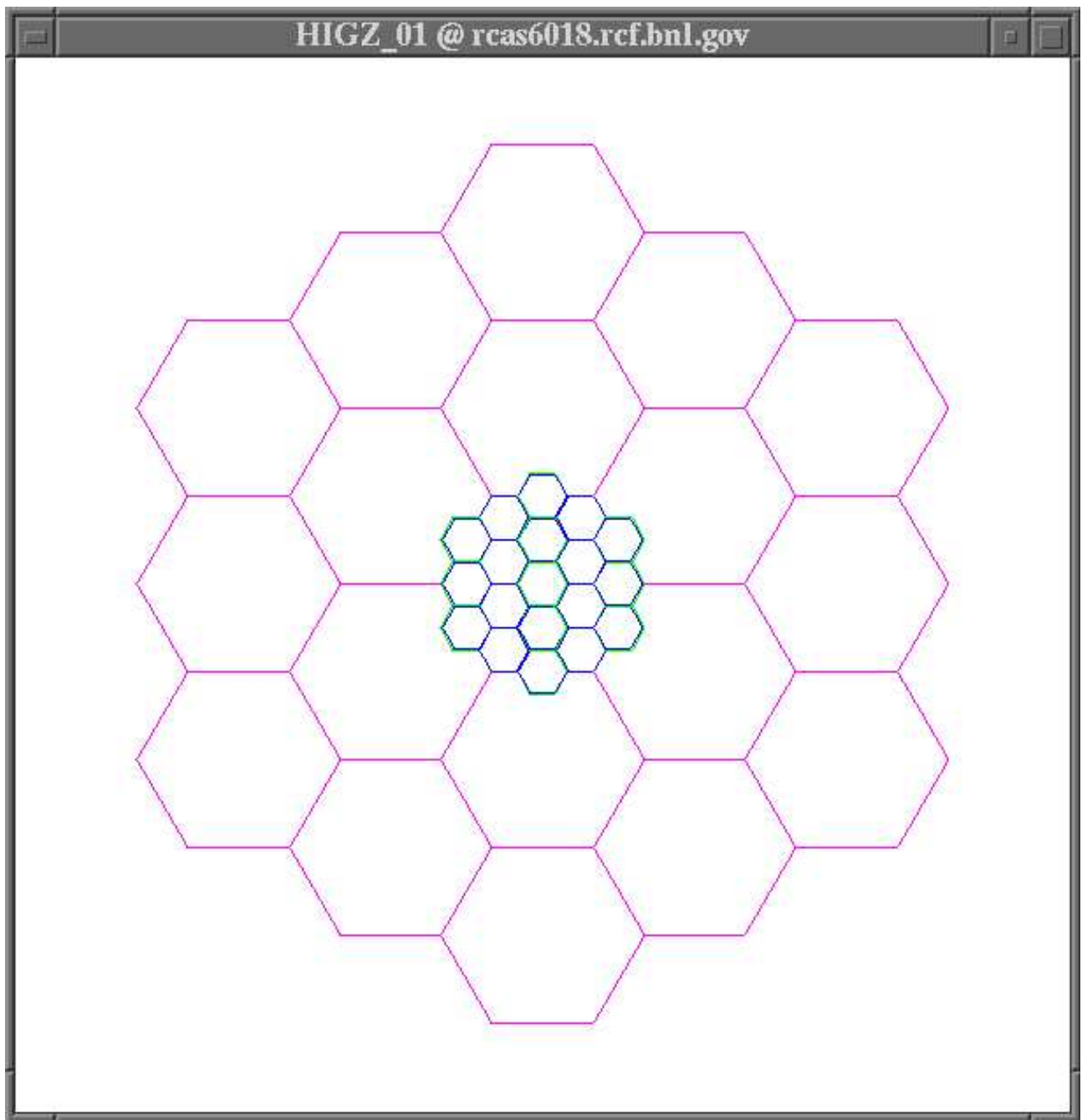


Looking at geometry in GSTAR:

```
staf > disp detm
```

Use on-screen controls to navigate among sub-systems, go to BBC:





```
staf > dcut BBCM Z 374.3 10 10 .1 .1
```



```
rxvt
is seen from the the same axis. If optional parameters are missing, the
current values in /GCDRAW/ are taken. When HIDE Mode is ON, it is possible
to get the same effect with the CVOL/BOX command.

staf > man dcut

Command "/GEANT/DRAWING/DCUT" :

* GEANT/DRAWING/DCUT NAME CAXIS CUTVAL [ UO VO SU SV ]

NAME      C 'Volume name'
CAXIS     C 'Axis value'
CUTVAL    R 'Cut plane distance from the origin along the axis'
UO        R 'U-coord. (horizontal) of volume origin'
VO        R 'V-coord. (vertical) of volume origin'
SU        R 'Scale factor for U-coord.'
SV        R 'Scale factor for V-coord.'

Possible CAXIS values are:

X
Y
Z

CALL GDRAWC(name,iaxis,cutval,u0,v0,su,sv)

The cut plane is normal to caxis (X,Y,Z), corresponding to iaxis (1,2,3),
and placed at the distance cutval from the origin. The resulting picture
is seen from the the same axis. If optional parameters are missing, the
current values in /GCDRAW/ are taken. When HIDE Mode is ON, it is possible
to get the same effect with the CVOL/BOX command.

staf > █
```

Use 'man' to learn more about dcut or any other command.

```
gstar_dAu.kumac
File Edit Options Buffers Tools Help

MACRO hijev nevent=8 file=hijing_dAu run=1
application data hijev.inp
-----
Hijing Control file
-----
Run number          : 1
Event number        : 0
Generator number    : 31
Frame/Energy        : 'CMS' 200.
Projectile type/A/Z : 'A' 2 1
Target type/A/Z     : 'A' 197 79
Impact parameter min/max (fm) : 0. 12.
Jet quenching (1=yes/0=no) : 0
Hard scattering/pt_jet (0/1, -thr) : 0 -2.25
Max # jets per nucleon (D=10) : 10
ihpr2(11), ihpr2(12)-pi0,k0,D,L,... decayoff : 1 1
ihpr2(21)- keep daughters, ihpr2(18) : 1 0
set C/B production(C=1.5,B=5.36) : 1.5
hijev.inp
* -----
* GSTAR setup
#detp geometry year_2a field=5
RNDM $pid [run]
vsig 0.01 30.
ghist [file].his
gstat time size mult stak
*
make hij
make geometry
    call bbcmeq; * recommended by Yiqun Wang to include BBC
gclose all
make gstar
make control
*
                               set a primitive dataset structure
mkdir evgen
cd evgen
tdm/newtable particle particle 40000
cd ..
*
                               I/O setup here
user/input      u  evgen/particle.staf
gfile           o  [file][run].fz
*
                               run event loop
do i = 1, [nevent]
    mod/call hijjet evgen/particle
    more          evgen/particle
    trig 1
enddo
*
exit
return
-----
-- gstar_dAu.kumac (Fundamental) --L22--C0--Top-----
Wrote /star/rcf/pwg/ebye/kopytin/trigger/trigger/dAu/gstar_dAu.kumac
```

HIJING events can be generated “on the fly” during the GSTAR session.

An alternative: "teach" the event generator to store events in a STAR-wide standard "HEPTUP" format, based on a Row-Wise Ntuple – done for popular generators by Ron Longacre.

\$STAR/pams/gen/heptup.f

implements the format.

```
* The following explanation was provided by Pavel Nevski:
* call HEPEvent
* (generator, run, Npart, B,F,Et,At, A1,Z1,A2,Z2 ) -new event
*   B -impact parameter
*   F -reaction plane angle
*   Et - energy
*   At - number of jets in HIJING, as used by Ron Longacre
* call HEPPart (ipa,ist,pdg, moth,idau,pp, Ep,Am,vv,vt)
* - write new particle
*   ipa - particle number, 1-npart
*   ist - status (stable - 1, unstable -2 )
*   pdg - Particle data book code (pythia & hijing use it)
*   moth(2) - numbers(ipa) of this particle's parents
*   idau(2) - numbers(ipa) -""- daughters
*   pp(3) - momentum px,py,pz
*   Ep - energy
*   Am - mass (particles may be off the mass shell)
*   momentum,energy and mass - in GeV
*   vv(3) - place of birth, vt - time of birth in mm and mm/c
*   can be simply 0*
* call HEPEnd (option) -close ntuple; compress on "z" option
```

Read the events file with

```
staf> user/input u evgen.1.nt
```



Output from GSTAR – I use FZ; it can be read and fed into the “Big Full Chain” for hit digitization.

Example of a shell script to do response simulation, based on GEANT hits:

```
#!/bin/tcsh
root4star -q -b 'bfc1.C(1,100000,\
"C2003 tfs fzin big mwc trg halfField EmcSim",\
"hijing_dAu1.fz","hijing_dAu1")'
# This will create file hijing_dAu1.event.root,
# subject to standard analysis
```



```

rxvt
* Starting   staf NwGEANT= 20000000 NwPAW= 2000000 *
*****pid= 16891*****
*****
*
*           W E L C O M E   to   staf
*
*           Version 2.00/04   7 October 2003
*
*****
Workstation type (?=HELP) <CR>=1 :
Version 1.29/03 of HIGZ started
1**** GEANT Version 3.21/14 Released on 19032002
0**** Correction Cradle Version 0.1400
**** Initialising Staf libraries
**** RZMAKE, OLD RZ format selected for RZDOC
staf > gfile p hijing_dAu1.fz
AGZOPEN opening file hijing_dAu1.fz
staf > trig
staf > gprint hits BBCH

====> HITS  IN DETECTOR ** BPOL ** OF SET ** BBCH ** <====

   HITS TRACK BBCM BBCA THXM SHXT      TOF      BIRK
   1     1     2     2     6     2  1.260E-08 1.689E-03
   2     1     2     2     6     3  1.249E-08 1.453E-03
   3     1     2     2     3     3  1.353E-08 1.937E-03
   4     1     2     2     3     3  1.437E-08 2.160E-04
   5     1     2     2     3     3  1.438E-08 9.193E-04
   6     1     2     2     3     3  1.440E-08 6.420E-04
   7     1     2     2     3     3  1.441E-08 6.392E-04
   8     1     2     2     3     3  1.443E-08 1.201E-03
   9     1     2     2     3     3  1.444E-08 1.395E-03
  10     1     2     2     3     3  1.446E-08 1.890E-03
  11     1     2     2     3     3  1.450E-08 2.254E-04
  12     1     2     2     3     3  1.743E-08 1.134E-03
  13     1     2     1     1     2  1.270E-08 1.030E-03
  14     1     2     1     1     2  1.272E-08 7.594E-04
  15     1     2     2     2     1  1.261E-08 1.209E-03
  16     1     2     2     2     1  1.263E-08 1.698E-03
  17     1     2     2     4     2  1.255E-08 4.585E-04
  18     1     2     2     4     2  1.257E-08 1.116E-03
  19     1     2     2     4     1  1.254E-08 2.798E-04
  20     1     2     2     4     1  1.257E-08 1.390E-03
  21     1     1     2     1     3  1.272E-08 1.606E-03
  22     1     1     2     3     3  1.603E-08 5.077E-04
  23     1     1     2     3     3  1.270E-08 1.710E-03
  24     1     1     2     1     3  1.255E-08 1.503E-03
  25     1     1     1     1     2  1.264E-08 1.441E-03
staf >

```

Need more cross-checks? read FZ file with events into STAF. Geometry is kept in the file (persistent) \Rightarrow have full access (can draw). Hits are saved. Use `trig` to move from event to event, `gprint` to print hits.



Steps needed for a full integration of a detector into GEANT and response simulation.

Step 1: Add the geometry and materials description: `pams/geometry/bbcmgeo/bbcmgeo.g`

Step 2: Select the appropriate hit type for the detector. For the BBC, it is a calorimetric hit with time-of-flight information; therefore, I used `g2t_ctf_hit`. `pams/sim/g2t/g2t_bbc.F` and `pams/sim/g2t/g2t_bbc.idl` are new files which define the structure of the hit.



Step 3: `g2t_volume_id.g` needs to be modified. This function returns and defines **volume ID** with a separate scheme for every subsystem; response simulation needs to know about the scheme and use it consistently when it makes decisions about volume elements. In GEANT, volume ID is encoded as an array representing the path to the volume, where array positions correspond to the levels of the hierarchical tree of volumes, and the array elements are IDs of the embedding volumes. Typically, a volume ID encodes this hierarchical information within a single integer number where different decimal places correspond to the levels of the tree. See example...

Return to [g2t_volume_id.g](#) CVS log

Up to [\[Offline\]](#) / [pams](#) / [sim](#) / [g2t](#)

Diff for /pams/sim/g2t/g2t_volume_id.g between version 1.37 and 1.38

version 1.37, 2001/09/06 00:22:00

version 1.38, 2002/10/16 19:12:44

Line 1

Line 1

```
* $Id$  
* $Log$
```

```
* $Id$  
* $Log$
```

```
* Revision 1.38 2002/10/16 19:12:44 kopytin  
* Volume ID for BBC elements added. If changed, will  
affect StBbcSimulationMaker.  
*
```

```
* Revision 1.37 2001/09/06 00:22:00 nevski  
* new svt geometry numbering introduced  
*
```

```
* Revision 1.37 2001/09/06 00:22:00 nevski  
* new svt geometry numbering introduced  
*
```

Line 530

Line 533

```
+ numbv(4)*100 + numbv(5)
```

```
+ numbv(4)*100 + numbv(5)
```

```
*16*
```

```
*16*
```

```
else if (Csys=='bbc') then  
* Mikhail Kopytine for the BBC group  
* BBC has 4 levels: west/east, annulus, triple module,  
single module  
volume_id = numbv(1)*1000 + numbv(2)*100 +  
numbv(3)*10 + numbv(4)  
*17*
```

```
else  
print *, ' G2T warning: volume ', Csys, ' not found '  
endif
```

```
else  
print *, ' G2T warning: volume ', Csys, ' not found '  
endif
```

Legend:

Removed from v.1.37

changed lines

Added in v.1.38

Colored Diff

Show



Step 4: In the ROOT/StMaker (C++) land: `St_geant_Maker.cxx` needs to be modified to read the hits from the new subsystem. **Need to know** the name of the hit (BBCH for BBC) and name of GEANT sensitive element (BPOL for BBC). The infrastructure generates a wrapper `g2t/St_g2t_bbc_Module.h`, based on `g2t_bbc.F` and `g2t_bbc.idl` – do not forget to include it!

Diff for /StRoot/St_geant_Maker/St_geant_Maker.cxx between version 1.73 and 1.74

version 1.73, 2002/08/26 14:21:21	version 1.74, 2002/10/16 20:39:23
Line 1	Line 1
<pre>// \$Id: St_geant_Maker.cxx,v 1.37 1999/04/19 06:29:30 nevski Exp // \$Id\$ // \$Log\$ // Revision 1.73 2002/08/26 14:21:21 jeromel // changing 'PHMD' to 'PHMH' // // Revision 1.72 2002/06/17 16:12:43 perev // fix wrong geant time</pre>	<pre>// \$Id: St_geant_Maker.cxx,v 1.37 1999/04/19 06:29:30 nevski Exp // \$Id\$ // \$Log\$ // Revision 1.74 2002/10/16 20:39:23 kopytin // Added code to read out BBC GSTAR tables. Needed // by StBbcSimulationMaker // // Revision 1.72 2002/06/17 16:12:43 perev // fix wrong geant time</pre>
Line 296	Line 296
<pre>#include "g2t/St_g2t_zdc_Module.h" #include "g2t/St_g2t_vpd_Module.h" #include "g2t/St_g2t_pmd_Module.h" #include "StarCallf77.h" #ifdef F77_NAME #define geometry F77_NAME(geometry,GEOMETRY)</pre>	<pre>#include "g2t/St_g2t_zdc_Module.h" #include "g2t/St_g2t_vpd_Module.h" #include "g2t/St_g2t_pmd_Module.h" #include "g2t/St_g2t_bbc_Module.h" #include "StarCallf77.h" #ifdef F77_NAME #define geometry F77_NAME(geometry,GEOMETRY)</pre>
Line 636	Line 640
<pre> iRes = g2t_zdc(g2t_track,g2t_zdc_hit); // ===== }</pre>	<pre> iRes = g2t_zdc(g2t_track,g2t_zdc_hit); // ===== geant3->Gfnhit("BBCH","BPOL", nhits); if (nhits>0) { St_g2t_ctf_hit *g2t_bbc_hit = new St_g2t_ctf_hit("g2t_bbc_hit",nhits); m_DataSet->Add(g2t_bbc_hit); iRes = g2t_bbc(g2t_track,g2t_bbc_hit); // ===== }</pre>
<pre>//-----all bloody detectors done-----//</pre>	<pre>//-----all bloody detectors done-----//</pre>



Step 5: retrieving data from a “table” in the digitization phase...

```
//  
Int_t StBbcSimulationMaker::Make()  
{  
  /// Make - this method is called in loop for each event  
  
  TDataSet* ds= GetInputDS("geant");  
  StEvent* event = (StEvent*)GetInputDS("StEvent");  
  St_g2t_ctf_hit* g2t_bbc_hit = (St_g2t_ctf_hit*)ds->Find("g2t_bbc_hit");  
  
  if (g2t_bbc_hit)  
  {  
    short nBBChits = g2t_bbc_hit->GetNRows();  
    BbcTOF TOFdata;  
    BbcDE DEdata;  
    for (short iBBChit=0; iBBChit<nBBChits; iBBChit++)  
    {  
      float De = g2t_bbc_hit->operator[] (iBBChit). de;  
      float TOF = g2t_bbc_hit->operator[] (iBBChit). tof;  
      short Vid = g2t_bbc_hit->operator[] (iBBChit). volume_id;  
  
      short PMTid = Geant2PMT[Vid]-1;  
  
      DEdata.AddDE (PMTid, De);  
      TOFdata.AddTOF (PMTid, TOF);  
    }  
  }  
}
```

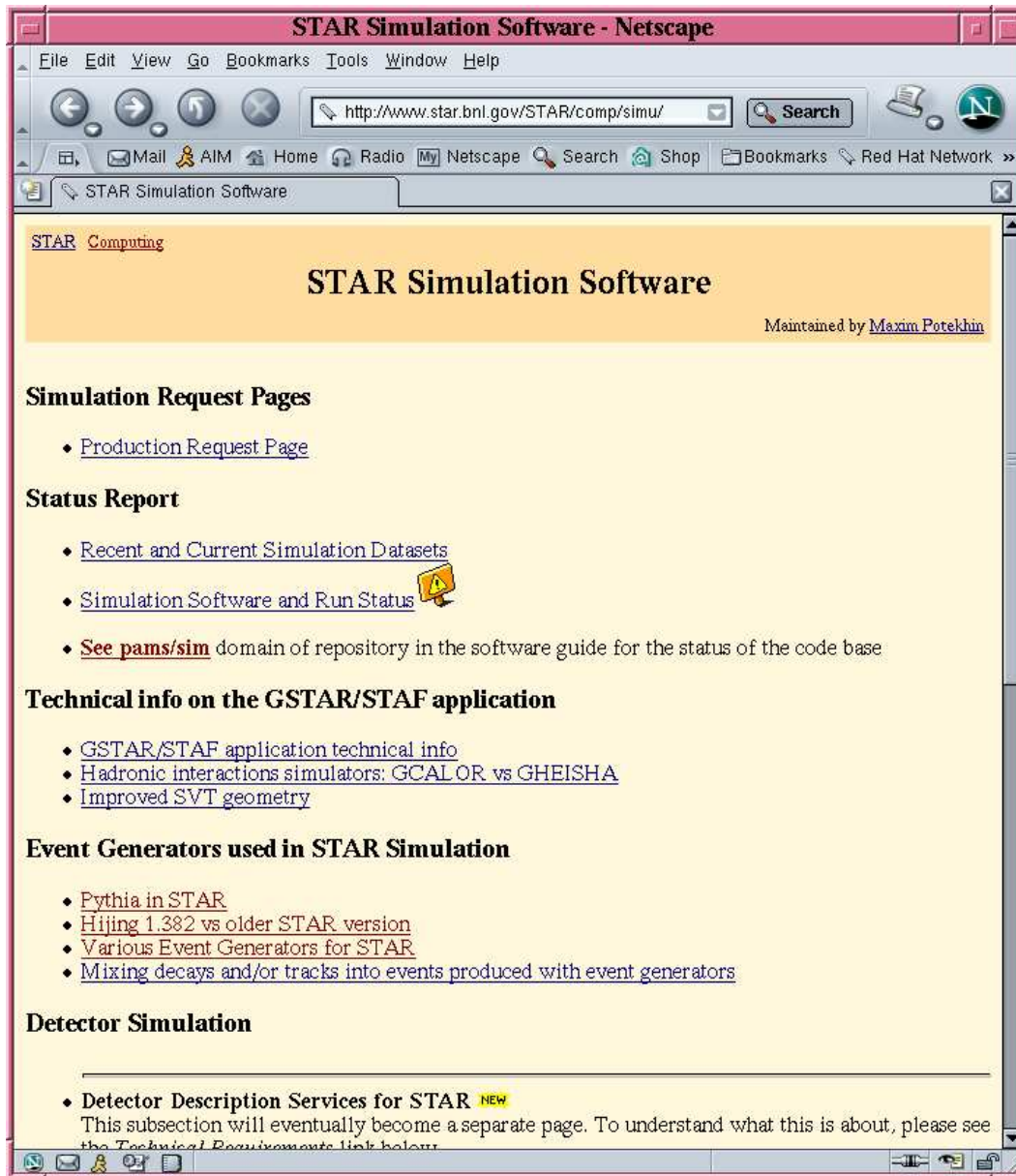
... and filling StEvent data structures (imitating calibrated output).

```
StTriggerDetectorCollection* myTrig = event->triggerDetectorCollection();
StBbcTriggerDetector& myBbc = myTrig->bbc();

for (u_short iPMT = 0; iPMT<NPMT2; iPMT++)
{
    short ADC = DEdata.GetADC(iPMT);
#ifdef BbcSimQa
    //      QaBbcPmtAdc->Fill(iPMT, ADC);
    short Vid = PMT2Geant[iPMT+1];

    if (Vid<2000) {QaBbcWestVid->Fill(ADC);}
    if (Vid>2000) {QaBbcEastVid->Fill(ADC);}
    if (iPMT<NPMT1) {QaBbcEastPmt->Fill(ADC);}
    if (NPMT1<=iPMT && iPMT<NPMT2) {QaBbcWestPmt->Fill(ADC);}
    QaBbcPmtTime->Fill(iPMT, TOFdata.GetTOF(iPMT));
    QaBbcPmtDE->Fill(iPMT, DEdata.GetDE(iPMT));
#endif
    myBbc.setAdc(iPMT, ADC);
    if (IsSmall(iPMT))
    {
        short TDC = TOFdata.GetTDC(iPMT);
        myBbc.setTdc(iPMT, TDC);
    }
}
else
{
    qMessMgr->Info
    ("MLK StBbcSimulationMaker::Make() could not inst g2t_bbc_hit\n");
}

return kStOK;
}
```



<http://www.star.bnl.gov/STAR/comp/simu/>

Credits: Pavel Nevski, Maxim Potekhin.