The STAR offline framework*

V. Fine

*) See also: **http://sol.star.bnl.gov/~fine/Publications/**Chep2000.ppt
http://chep2000.pd.infn.it/abs/abs_326.htm
http://root.cern.ch/root/R2000Abstracts.html

BROOKHAVEN NATIONAL LABORATORY

STAR reconstruction framework*

* "*… a set of cooperating classes that make up a reusable design for a specific class of software ...*"
by **Erich Gamma**,et al.
"Design Patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley Pub Co, 1995.

"The framework dictates the architecture of your application. It defines the over-all structure, its partitioning into classes and objects, the key responsibilities thereof, how the classes and objects collaborate, and the thread of control. "

A framework predefines these design parameters so **physicists can** design their solutions using a proven programming model and can **concentrate on the specifics of their applications**.

STAR framework is designed to support the chained components, which can themselves be composite sub-chains, with components ("*makers*") managing "*datasets*" they have created and are responsible for.

An **St_DataSet** class from which data sets and makers inherit allows the construction of hierarchical organizations of components and data, and centralizes almost all system tasks:

- **data set navigation**,
- **I/O, database access**,
- **inter-component communication.**

Permanent stable interface - various datas

We do not know how our data will look like,
but we know how we will access them

Abstract interface, hidden data approach

Various interfaces - permanent data

We do know how our data will look like,
but we do not know how we will access them

Dataset model, transparent data

Basic St_DataSet properites

St_DataSet object ::= the "named" collection of St_DataSet objects

•***Dataset Member***. Any object from the collection above is called "***DataSet Member***"

•***Structural member***. The "*Dataset Member*" is its "*Structural member*" if its "back pointer" points to this object

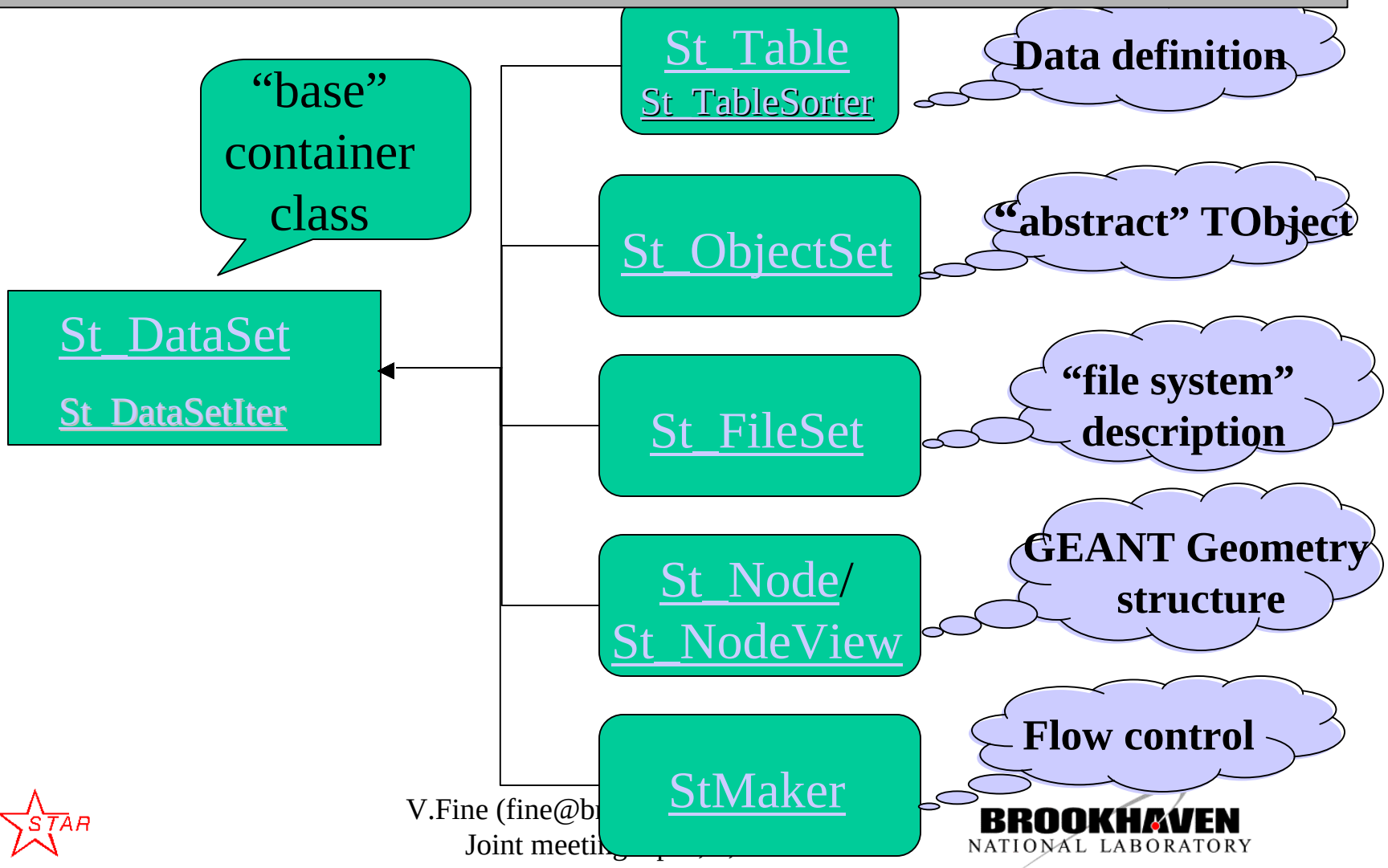
•***Dataset Owner*** (parent). We will say this St_DataSet object "owns" (or is an owner / parent of) another St_DataSet object if the last one is its "*Structural Member*"

• ***Associated member***. If some object is not "*Structural member*" of this object we will say it is an "*Associated Member*" of this dataset

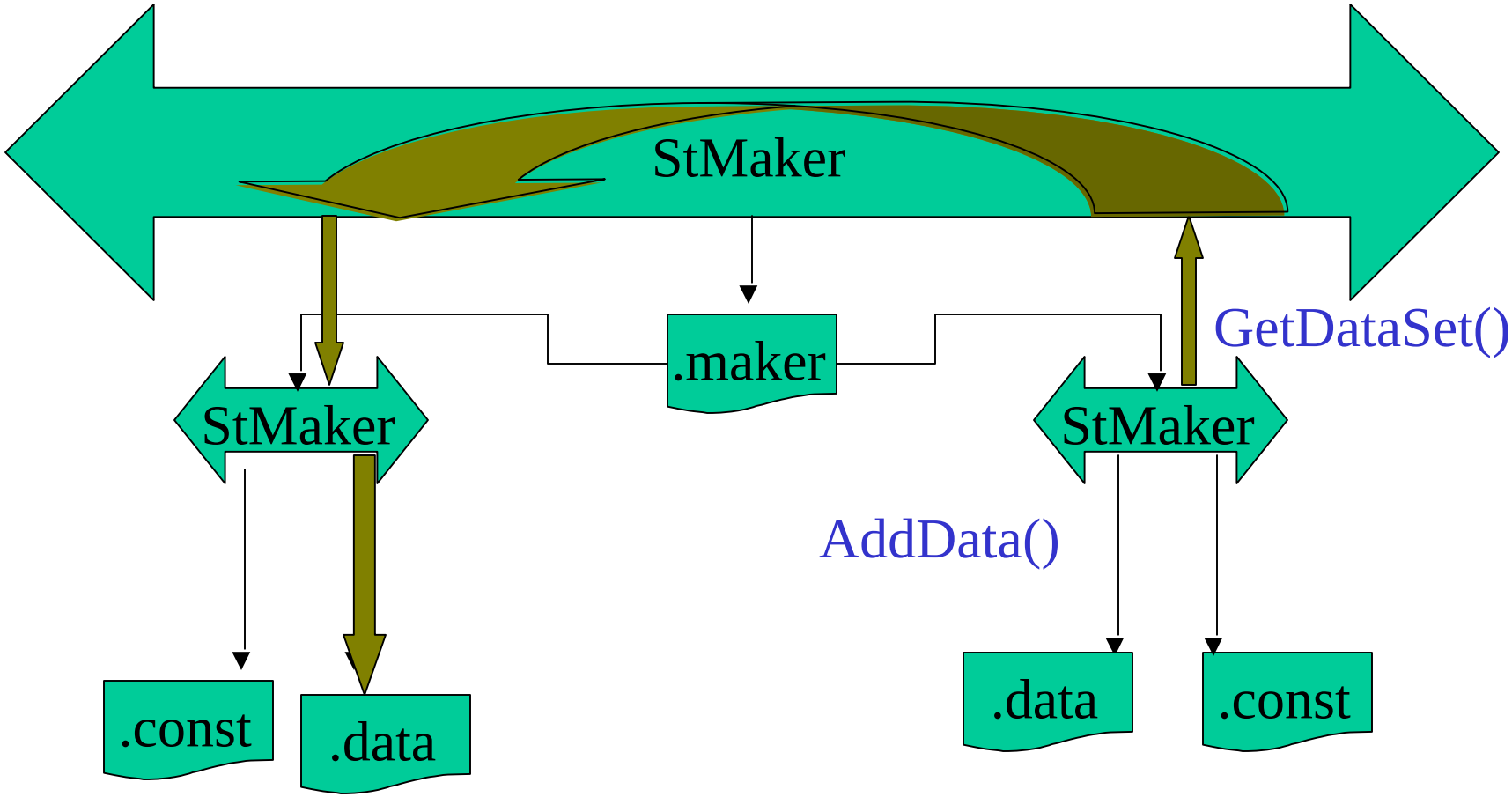
• ***Orphan dataset***. If some dataset is a member of NO other St_DataSet object it is called an "*orphan*" dataset object

OO model of the STAR simulation / reconstruction chain:

St_DataSet object ::= the "named" collection of St_DataSet objects



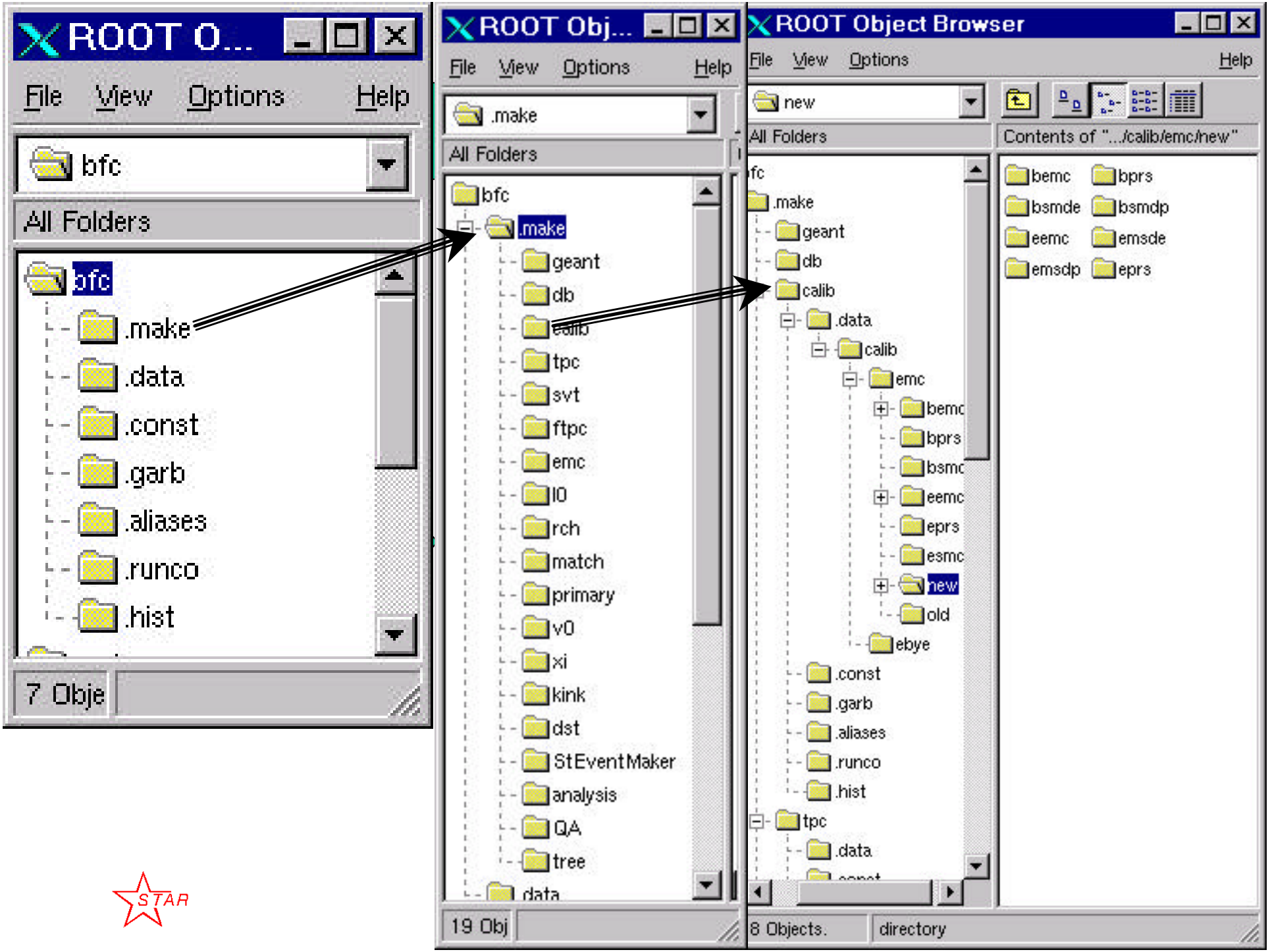
BROOKHAVEN NATIONAL LABORATORY



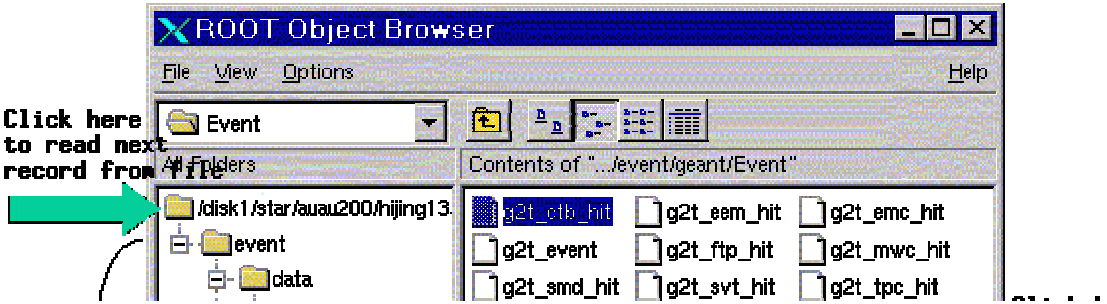
1. Init()
2. Make()

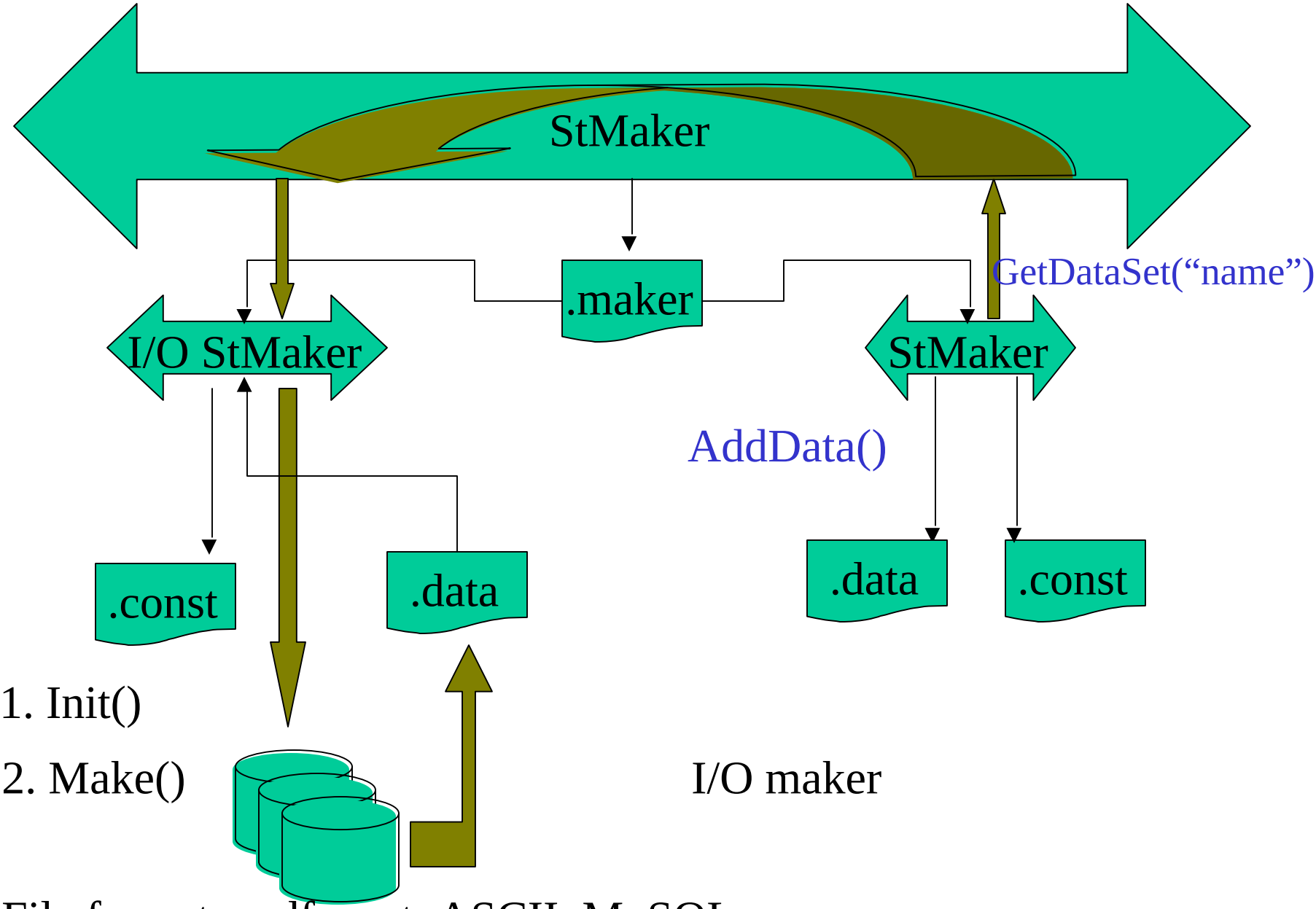
“regular” makers communication

No transaction changes the dataset relationship


ROOT O...
File View Options Help
bfc
All Folders
bfc
.make
.data
.const
.garb
.aliases
.runco
.hist
7 Obje
ROOT Obj...
File View Options Help
.make
All Folders
bfc
.make
geant
db
calib
tpc
svt
ftpc
emc
l0
rch
match
primary
v0
xi
kink
dst
StEventMaker
analysis
QA
tree
data
19 Obj
ROOT Object Browser
File View Options Help
new
All Folders
Contents of ".../calib/emc/new"
fc
.make
geant
db
calib
.data
calib
emc
bemc
bprs
bsmc
eemc
eprs
esmc
new
old
ebye
.const
.garb
.aliases
.runco
.hist
tpc
.data
bemc
bprs
bsmde
bsmdp
eemc
emsde
emsdp
eprs
8 Objects.
directory
STAR

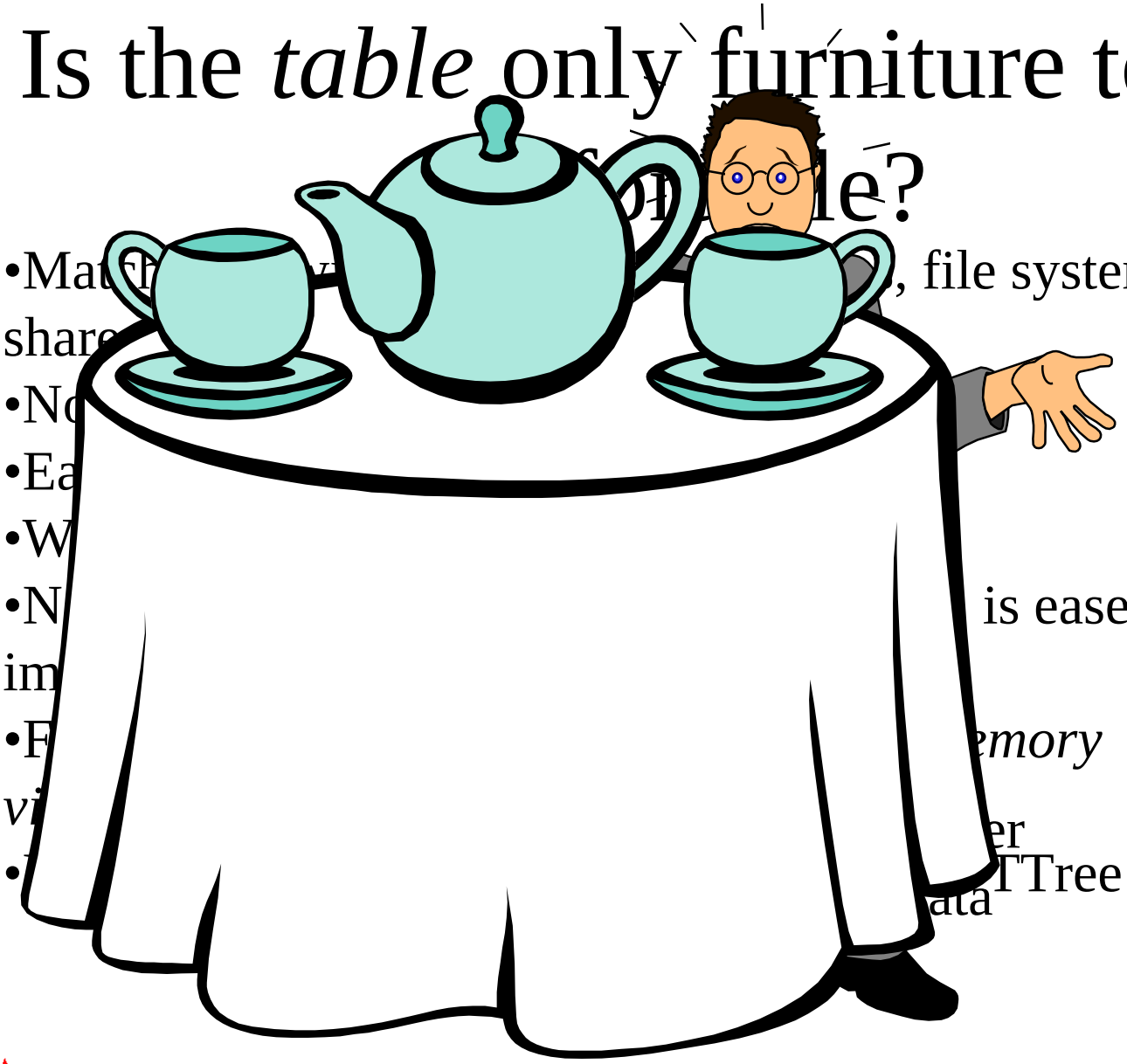
Typical STAR St_DataSet/St_Table structure





File formats: xdf, root, ASCII, MySQL

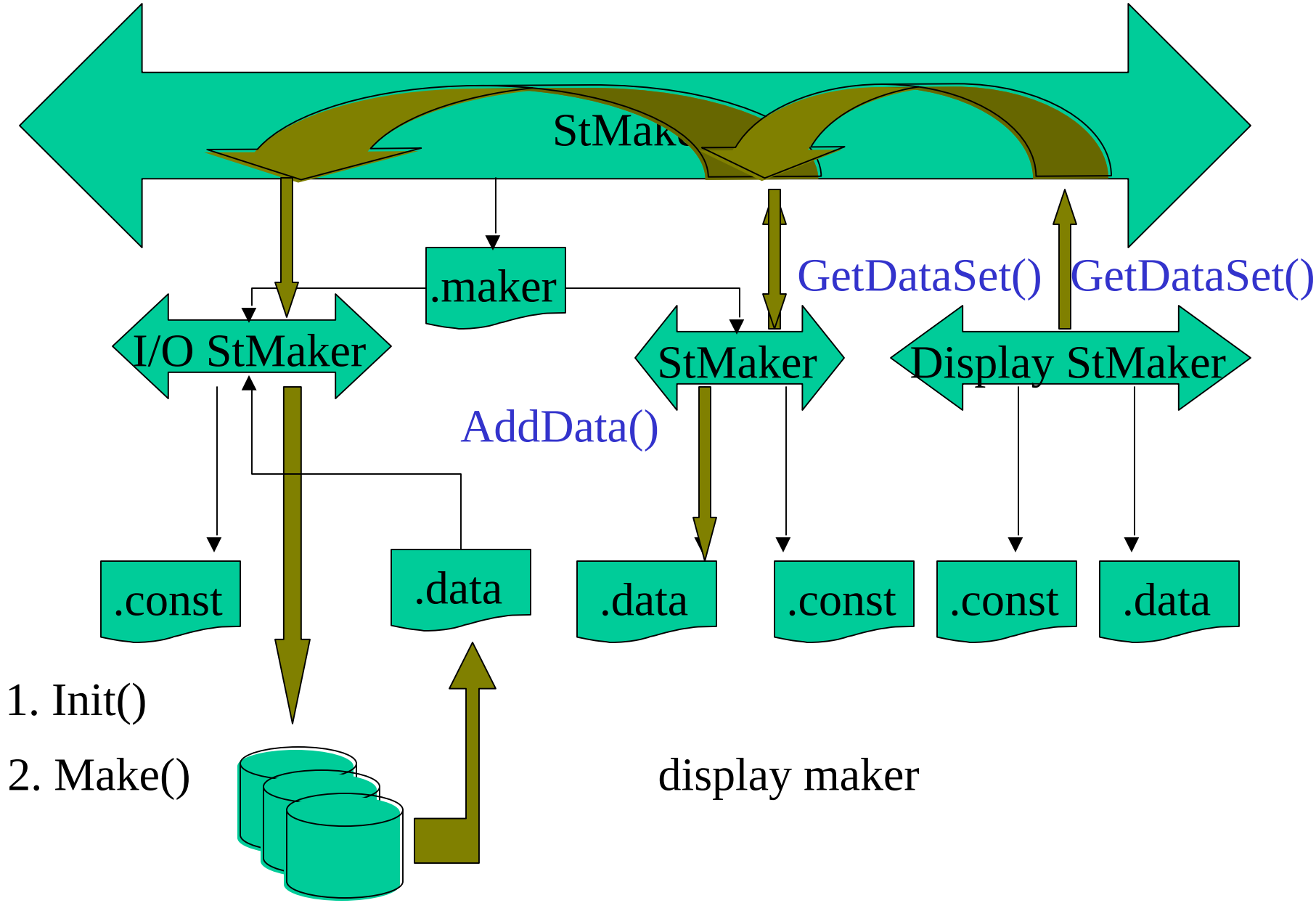
Is the *table* only furniture to be



One needs “chairs”

• **`TVolume`** class to define GEANT-like detector geometry. This class gives an access to the full detector definition from GEANT for visualization and coordinate transformation.

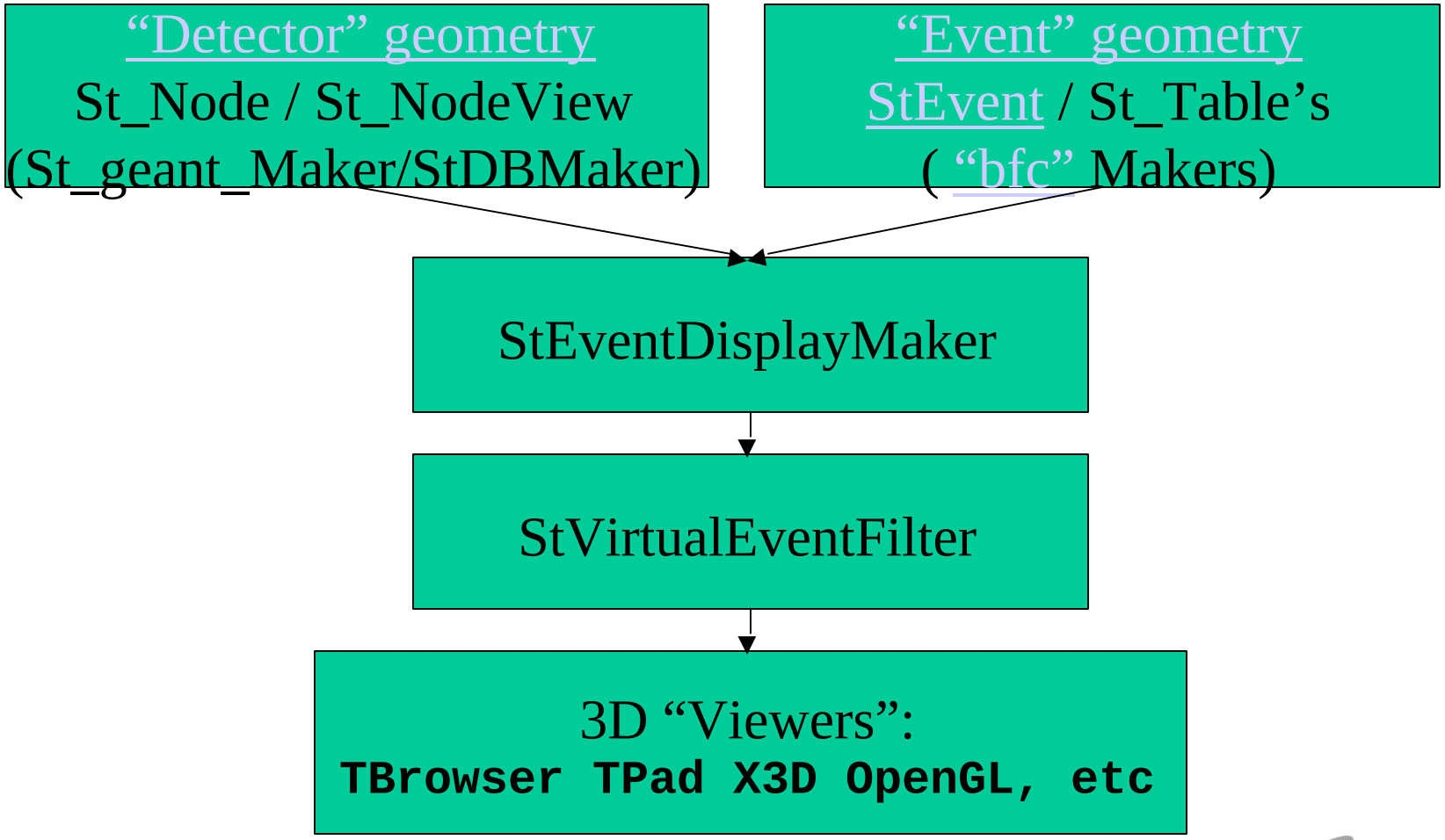
•“Proxy” base class - **`TChair.`** To get his / her custom access to one and the same table the USER may derive the class from TChair. Every individual or working group can create as many chairs as they find useful with no argue which chair is better.


StMaker
.maker
GetDataSet()
GetDataSet()
I/O StMaker
StMaker
Display StMaker
AddData()
.const
.data
.data
.const
.const
.data
1. Init()
2. Make()
display maker

OO model of STAR geometry

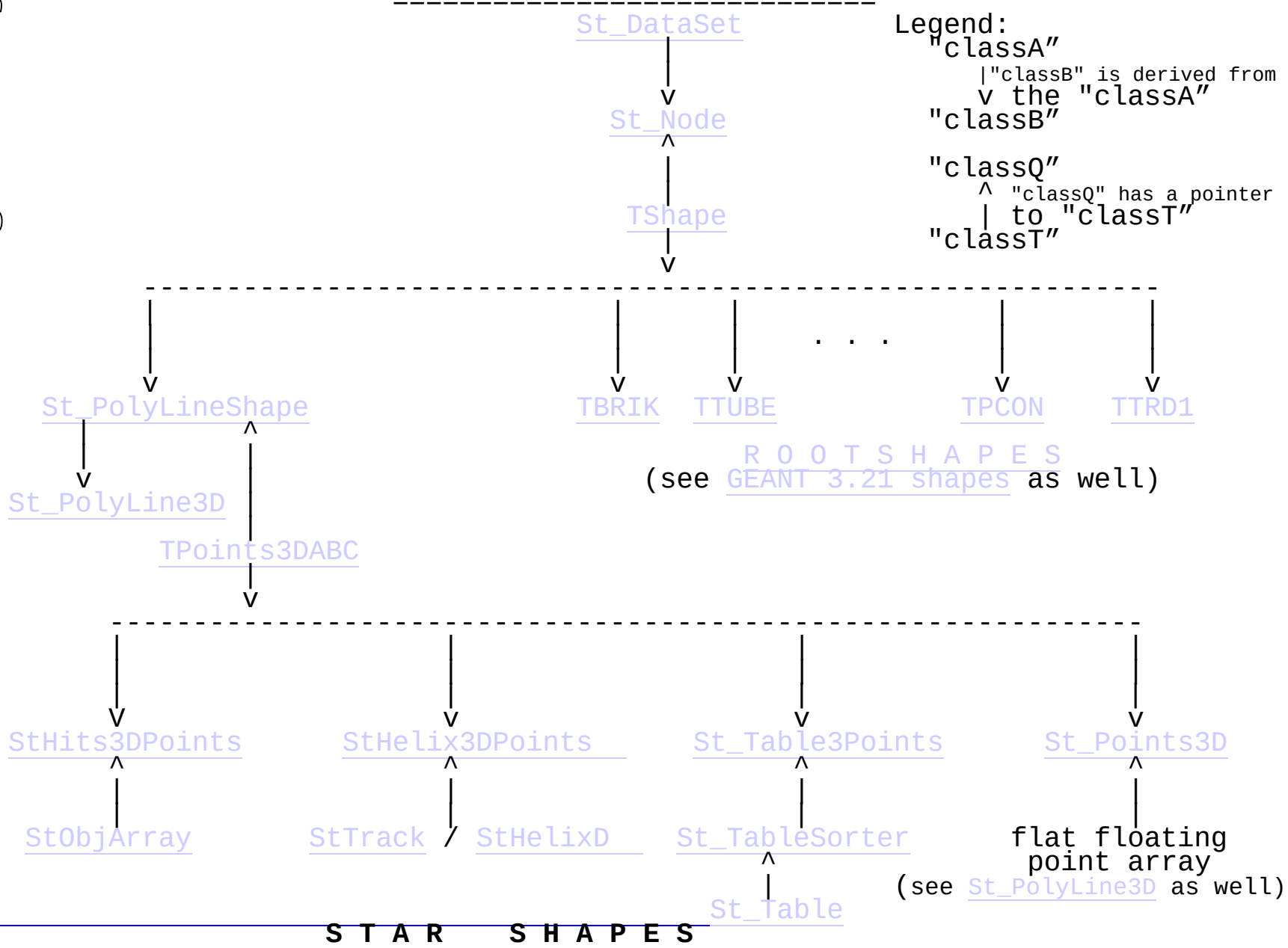
- **"Detector" geometry supplied by St_geant_Maker (GEANT 3.21)**
- **"Event" geometry supplied by "bfc.C" Makers of the reconstruction chain.**

StEventDisplayMaker



Event geometry Detector geometry

STAR 3D geometry Object Model

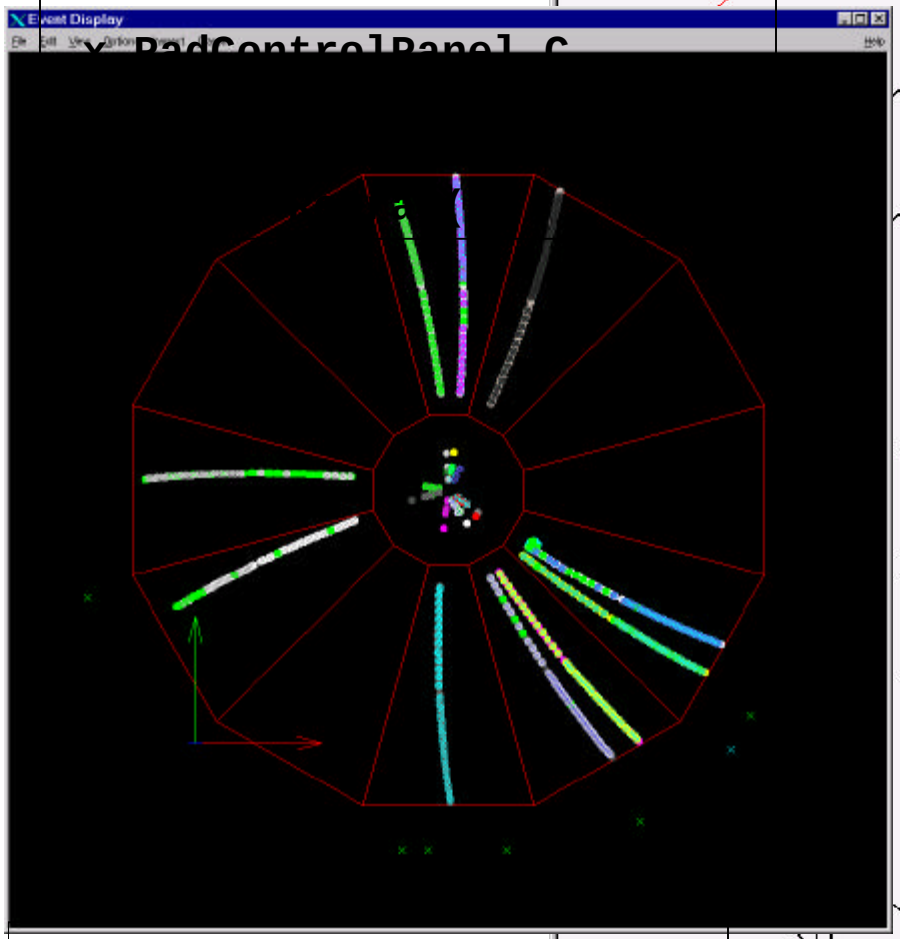


S T A R S H A P E S

V.Fine (fine@bnl.gov) ALICE-STAR
Joint meeting April, 9, 2000

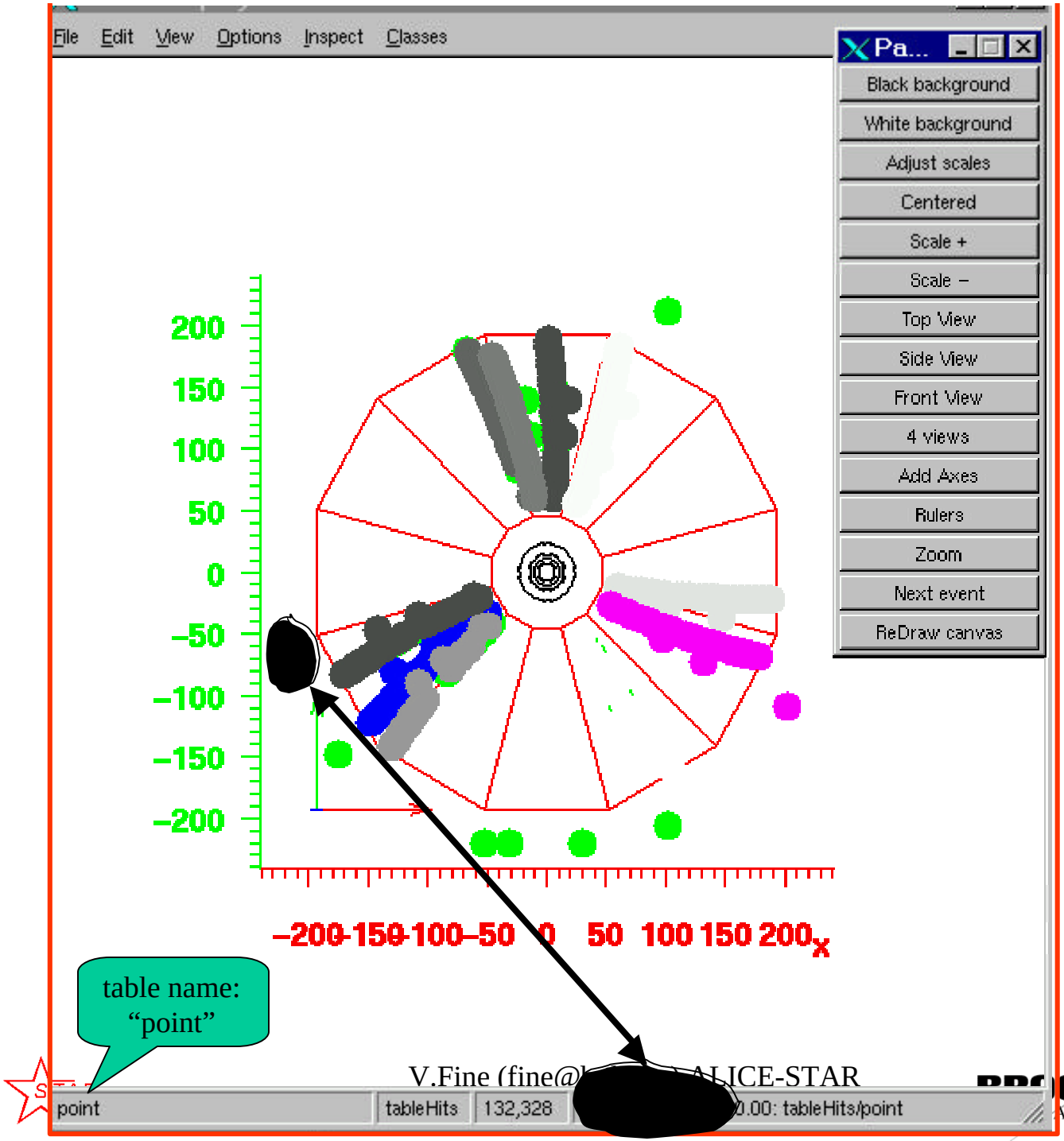
BROOKHAVEN NATIONAL LABORATORY

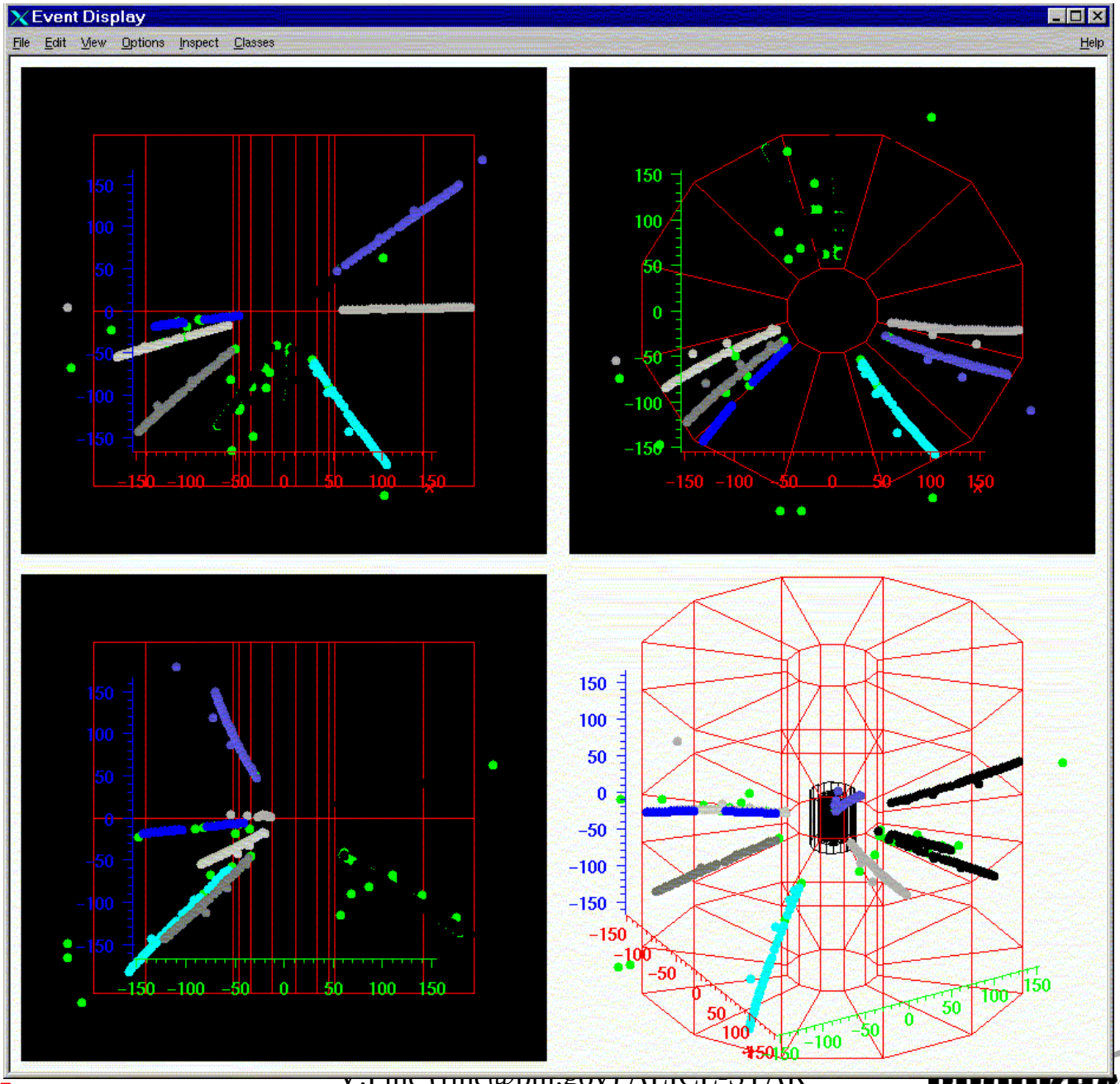
```
.x bfc.C(2)
.x TurnDisplay.C("no")
```

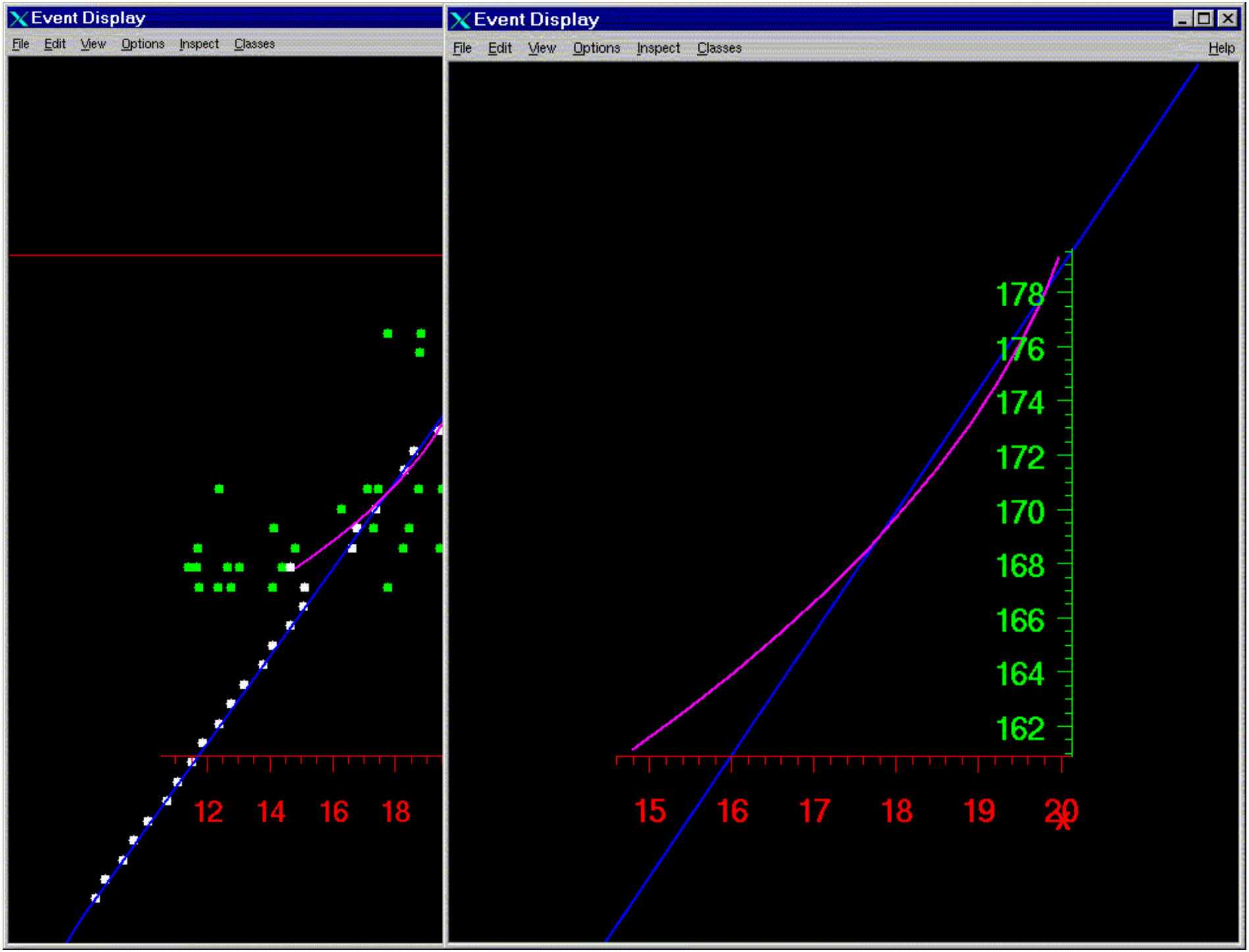


These pictures present "**g2t_svt_hit**" (round filled mark) vs packed "**dst_point**" (cross mark).

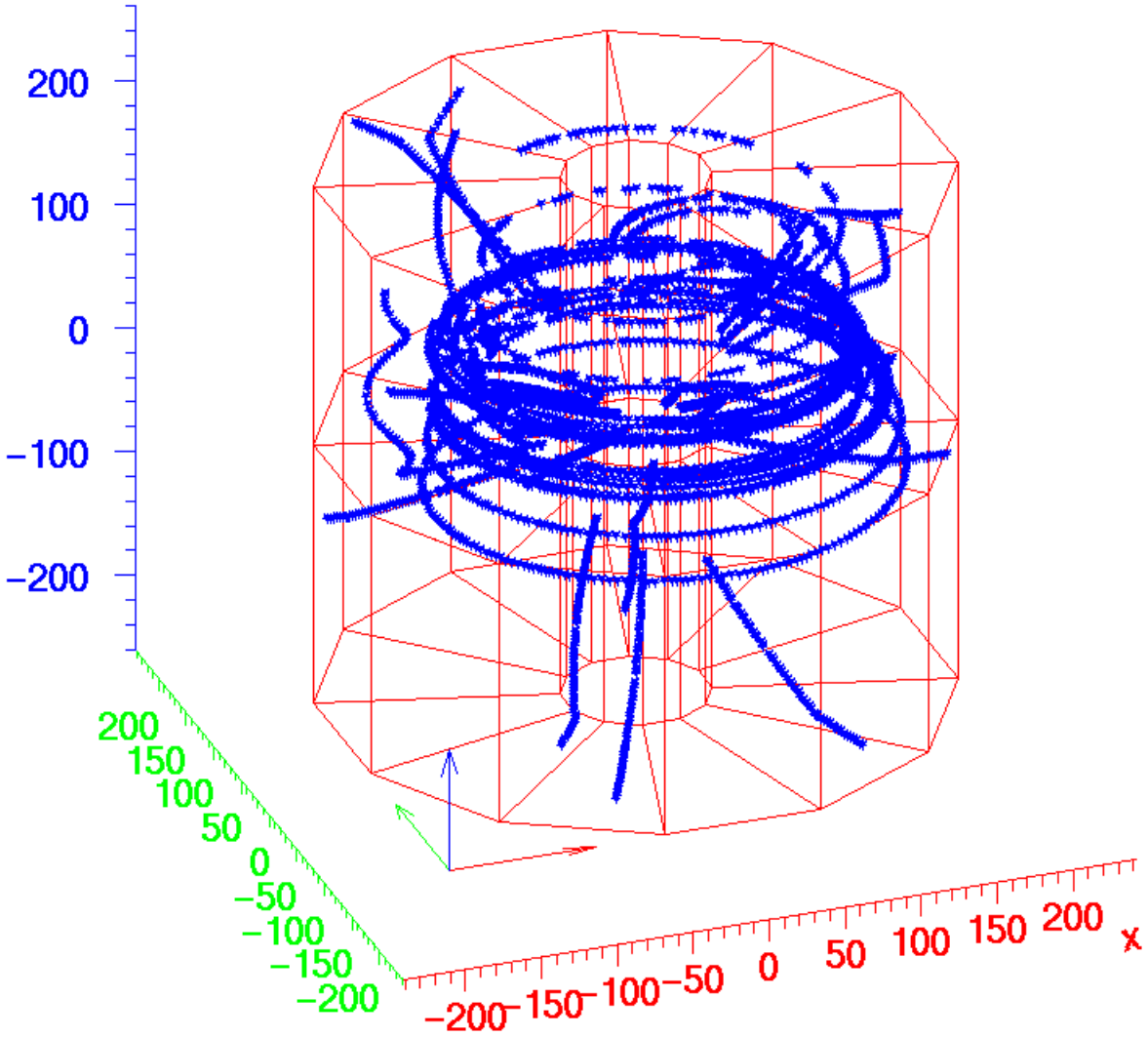
It is clear the hits on the outermost **svt** layer from **g2t** are not included into the "packed" **dst_point** table.

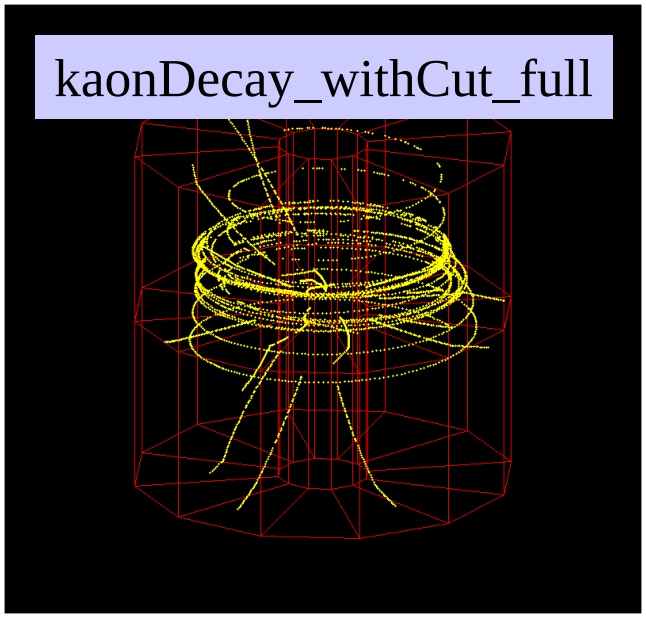

File Edit View Options Inspect Classes
Pa...
Black background
White background
Adjust scales
Centered
Scale +
Scale –
Top View
Side View
Front View
4 views
Add Axes
Rulers
Zoom
Next event
ReDraw canvas
200
150
100
50
0
–50
–100
–150
–200
–200 –150 –100 –50 0 50 100 150 200 x
table name:
“point”
point
tableHits
132,328
0.00: tableHits/point


Event Display
File Edit View Options Inspect Classes Help


Event Display
File Edit View Options Inspect Classes
12 14 16 18
Event Display
File Edit View Options Inspect Classes Help
178 176 174 172 170 168 166 164 162
15 16 17 18 19 20

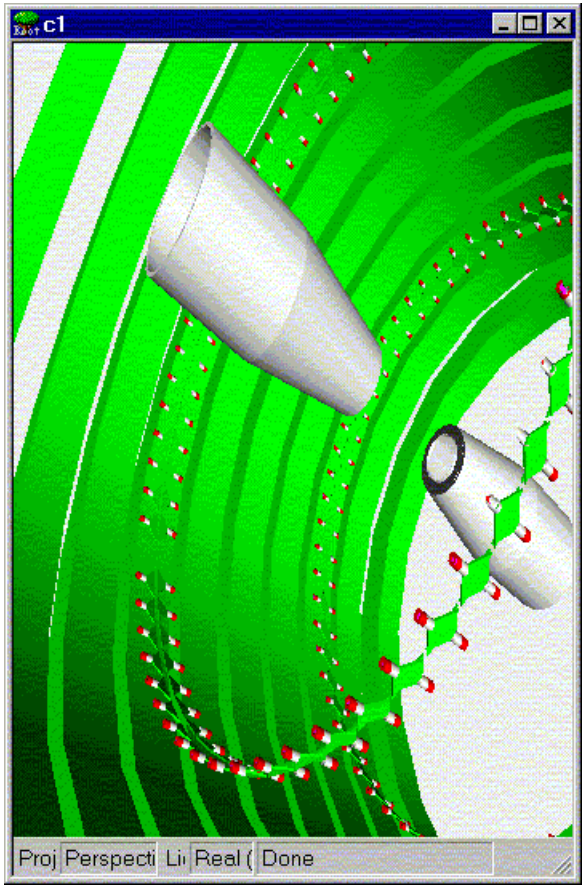
```
.x bfc.C(1,"off tpc tfs y1b eval fzin global -v0 -xi",
        "/disk0/star/test/venus412/b0_3/year_1b/set0352_01_35evts.fzd")
.x TurnDisplay.C("StTrackFilter2")
.x SetObjectFilter.C
.x PadControlPanel.C
```

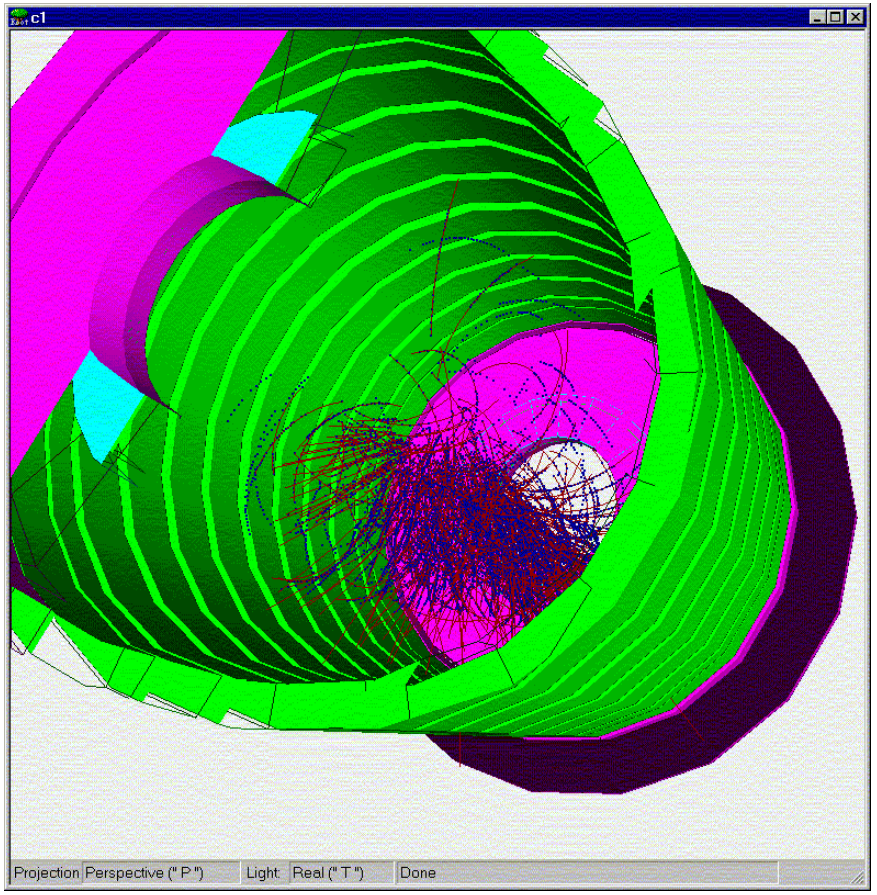




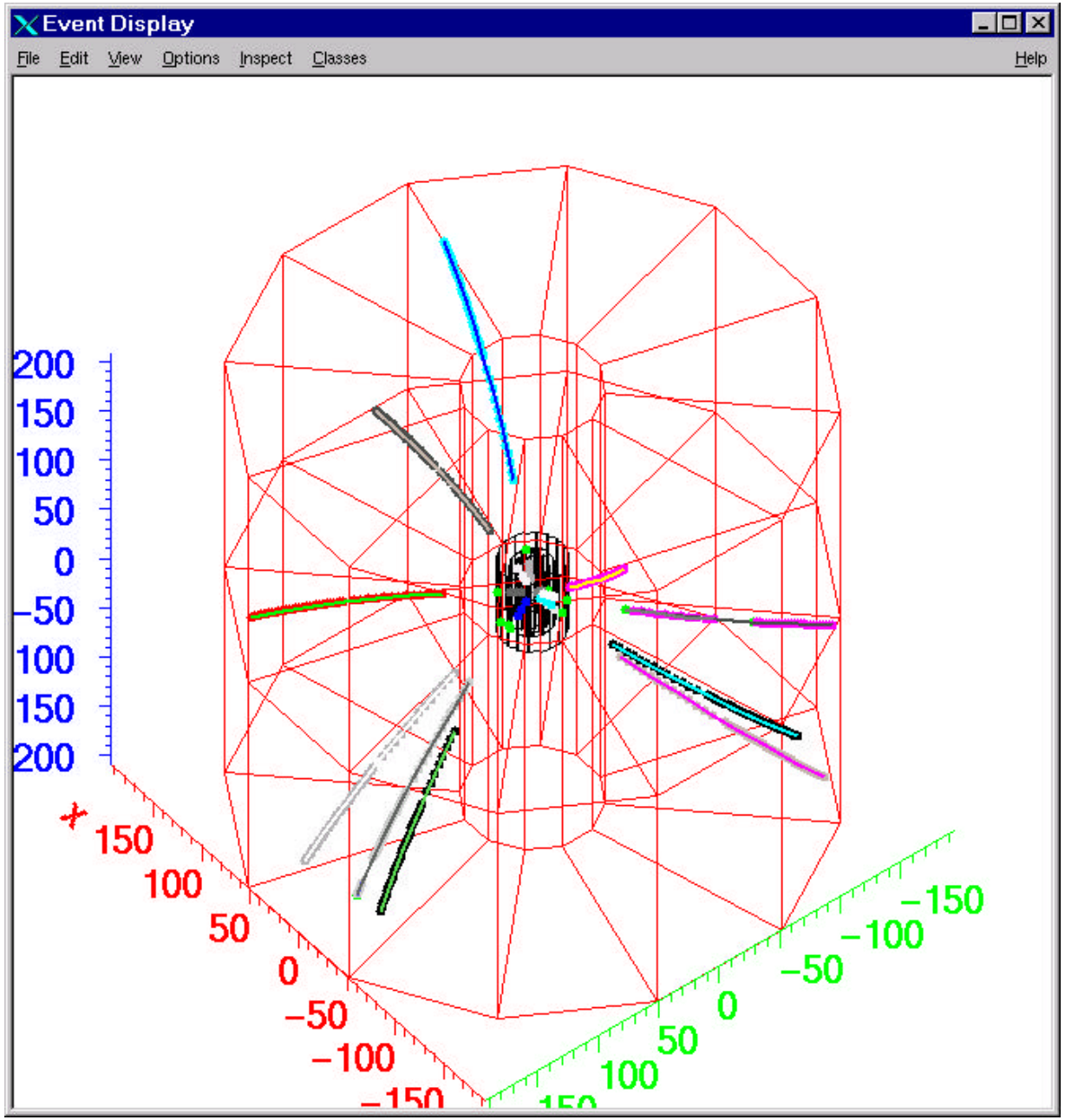
BROOKHAVEN NATIONAL LABORATORY

OpenGL viewer





BROOKHAVEN NATIONAL LABORATORY


Event Display
File Edit View Options Inspect Classes Help
200
150
100
50
0
−50
100
150
200
150
100
50
0
−50
−100
−150
−150
−100
−50
0
50
100

Current status

Language	CHEP'98	CHEP'2000	2000/98
C++	**27**	**138**	**5.1**
FORTRAN	**90**	**68**	**0.75**
Mortran	28	34	1.2
C	20	24	1.2
IDL	10	13	1.3
ROOT macros		11	
KUIP	22	4	0.2
Total klocs	**197**	**292**	**1.5**

The present framework has being testing for the last year. It was used to produce 100 GBytes of DST from 3 TBytes of the GEANT-simulated data. It is proved it allows the construction of hierarchical organizations of components and data, and centralizes almost all system tasks such as data set navigation, I/O, database access, and inter-component communication.