

Reactor
Analysis
Tool:
Tutorial

From zero to 60 plots/hour in an afternoon!

Stan Seibert

University of Texas

October 27, 2005



Our Mission:

Plot the energy distribution for
inverse beta-decay events.

Plan of Attack

1. Install Prerequisites
2. Download and Compile
3. Run Tests
4. Create Command File
5. Run RAT
6. Make Plots

Install Prerequisites

- Operating systems:
Sun Solaris 8, RedHat Linux 7.3,
Scientific Linux 4, Mac OS X 10.4 (most
any recent Linux distro should also work)
- CLHEP 1.9.2.1
- GEANT 4.7.1
 - “Copy all headers to one directory”
 - “Make static libraries”
- ROOT 4.04.02g
 - no ROOT 5 yet

**Version
numbers are
important!**

Download and Compile

Check out from CVS:

```
cvs -d:pserver:[username]@cp4.uchicago.edu:/theta13/cvs co RAT
```

Run configure script:

```
./configure
```

Setup Environment:

```
source env.sh
```

Compile:

```
make
```

Only need to do this once!



Install Gd neutron capture data

```
make installdata
```

Run Tests: *“Trust, but verify”*

RAT includes a (*non-comprehensive*) test suite.
Good for checking things on a new platform:

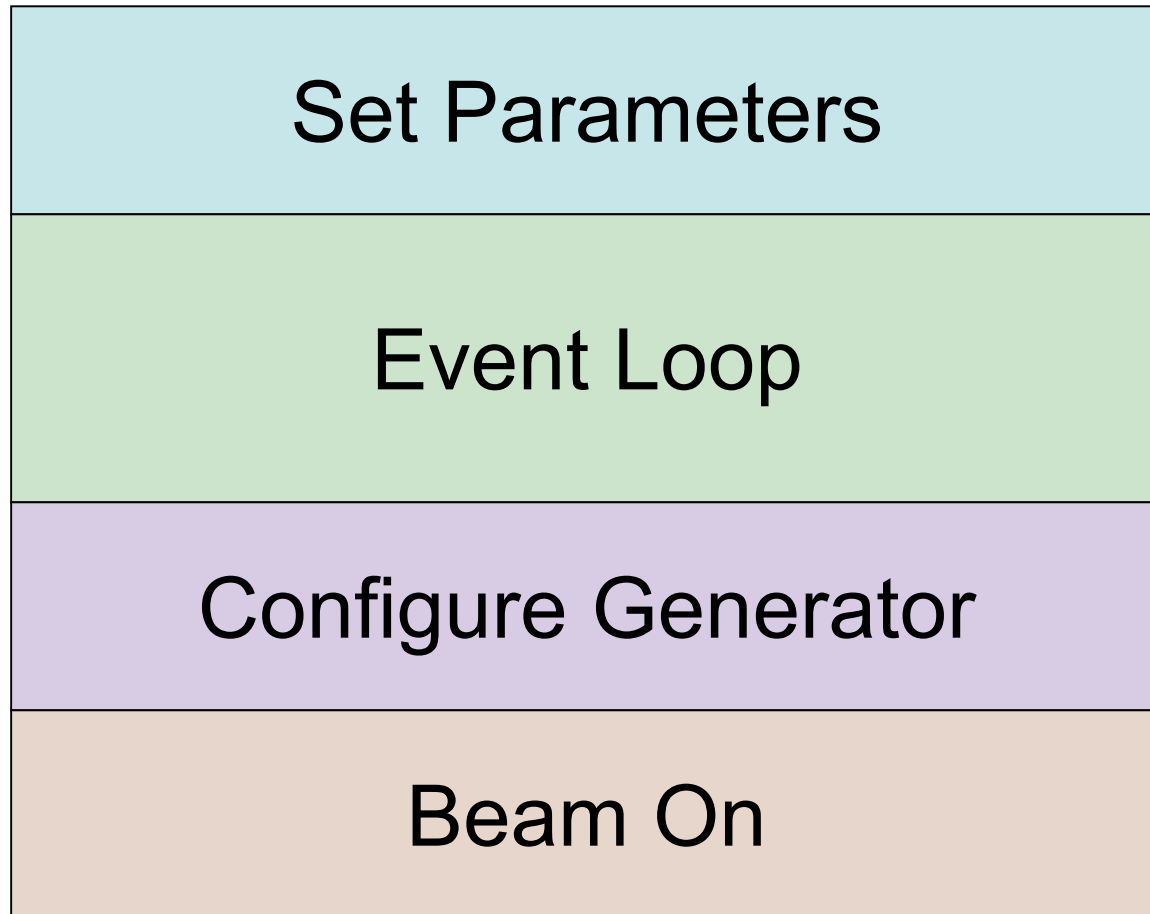
```
Running 54 testsRATDB: Loading /data/bw/RAT/data/IBD.ratdb ...  
.....RATDB: Loading /data/bw/RAT/data/IBD.ratdb ... Success  
..  
In TestRATDB::testLoad:  
TestRATDB.hh:25: Trace: Should see error message next.  
RATDB: Error! Cannot open noexist.ratdb  
.RATDB: Loading ./test.ratdb ... Success!  
RATDB: Loading ./test2.ratdb ... Success!  
.....RATDB: Setting TEST[1] myint to 16  
RATDB: Setting TEST[1] myfloat to 1.6  
RATDB: Setting TEST[1] mydouble to 16.5d-3  
RATDB: Setting TEST[1] mystring to ratratrat  
.RATDB: Setting TEST[1] myint to 16  
RATDB: Setting TEST[1] myfloat to 1.6  
RATDB: Setting TEST[1] mydouble to 16.5d-3  
RATDB: Setting TEST[1] mystring to ratratrat  
.You should see this warning  
You should see this info line  
.You should see this warning  
You should see this info line  
.....OK!
```

```
cd test  
make test
```



Creating the Command File

Command files have several major sections:



Command File: Set Parameters

- Can disable muon and hadron physics to speed up simulation.
- We want hadron processes on, for neutron capture, and muon processes off:

```
/glg4debug/glg4param omit_muon_processes 1.0  
/glg4debug/glg4param omit_hadronic_processes 0.0  
  
/run/initialize
```



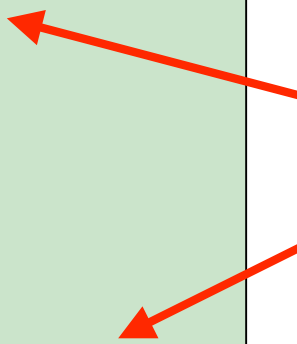
Sets up physics tables in GEANT4

Command File: Event Loop

- What processors do we want?
 - Status messages, fitting, output to ROOT file

```
/rat/proc count  
/rat/procset update 1  
  
/rat/proc fitpoisson  
  
/rat/proc outroot  
/rat/procset file "ibd.root"
```

*Processors are
configured with the
/rat/procset
command*



Command File: Configure Generator

- Inverse beta decay generator
- Position: center
- Incident neutrino along x-axis
- Currently requires “magic numbers”, but soon that will not be required

```
/process/inactivate DeferTrackProc
/generator/event_window 10000000

/generator/rates 52 0.001
/generator/pos/set 0 "0 0 0"
/generator/vtx/set 20 "1. 0. 0."
```

Neutrons
take a long
time to
capture

IBD in scint

Pos gen in scintillator

IBD vertex gen

Command File: Beam On

- Simulation starts as soon as beam on command is reached!

```
/run/beamOn 100
```

Command File: All Together

```
/glg4debug/glg4param omit_muon_processes 1.0
/rlg4debug/glg4param omit_hadronic_processes 0.0
/run/initialize

/rat/proc count
/rat/procset update 1
/rat/proc fitpoisson
/rat/proc outroot
/rat/procset file "IBD.root"

/process/inactivate DeferTrackProc
/generator/event_window 10000000
/generator/rates 52 0.001
/generator/pos/set 0 "0 0 0"
/generator/vtx/set 20 "1. 0. 0."

/run/beamOn 100
```

Running RAT



Execute macro by running the rat program:

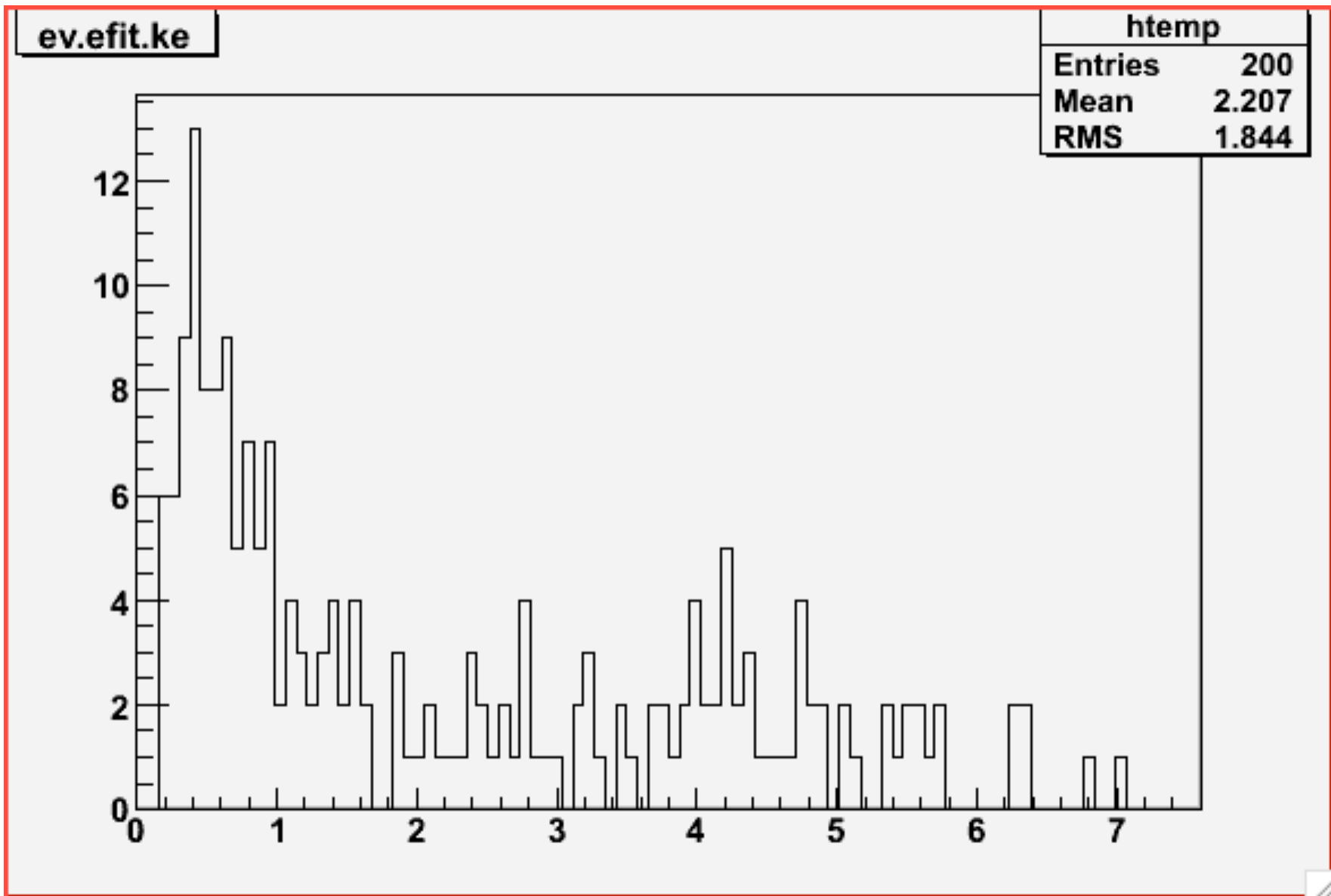
```
rat ibd.mac
```

RAT creates a log file called `rat.log` by default. All output to the screen is also written to the file. Change the name with a command line option:

```
rat -l ibd.log ibd.mac
```

Make Plots

- By default, ROOT file contains entire data structure.
- Starting ROOT in RAT/ directory will automatically load the shared library with data structure classes.
- TTree object is called T , and data structure is on the `ds` branch.
- See User's Manual (or TBrowser) for full list of data structure members.



```
T->Draw("ev.efit.ke");
```

For more information:

- Example command files in RAT/mac/
- The RAT User's Manual:
<http://nu.ph.utexas.edu/bw/trac/wiki/UserGuide>
 - Installation guide
 - Full list of event data structure members
 - Processor documentation
- GLG4sim Homepage:
<http://neutrino.phys.ksu.edu/~GLG4sim/>
- Email the list:
`bw_sim@hep.chicago.edu`