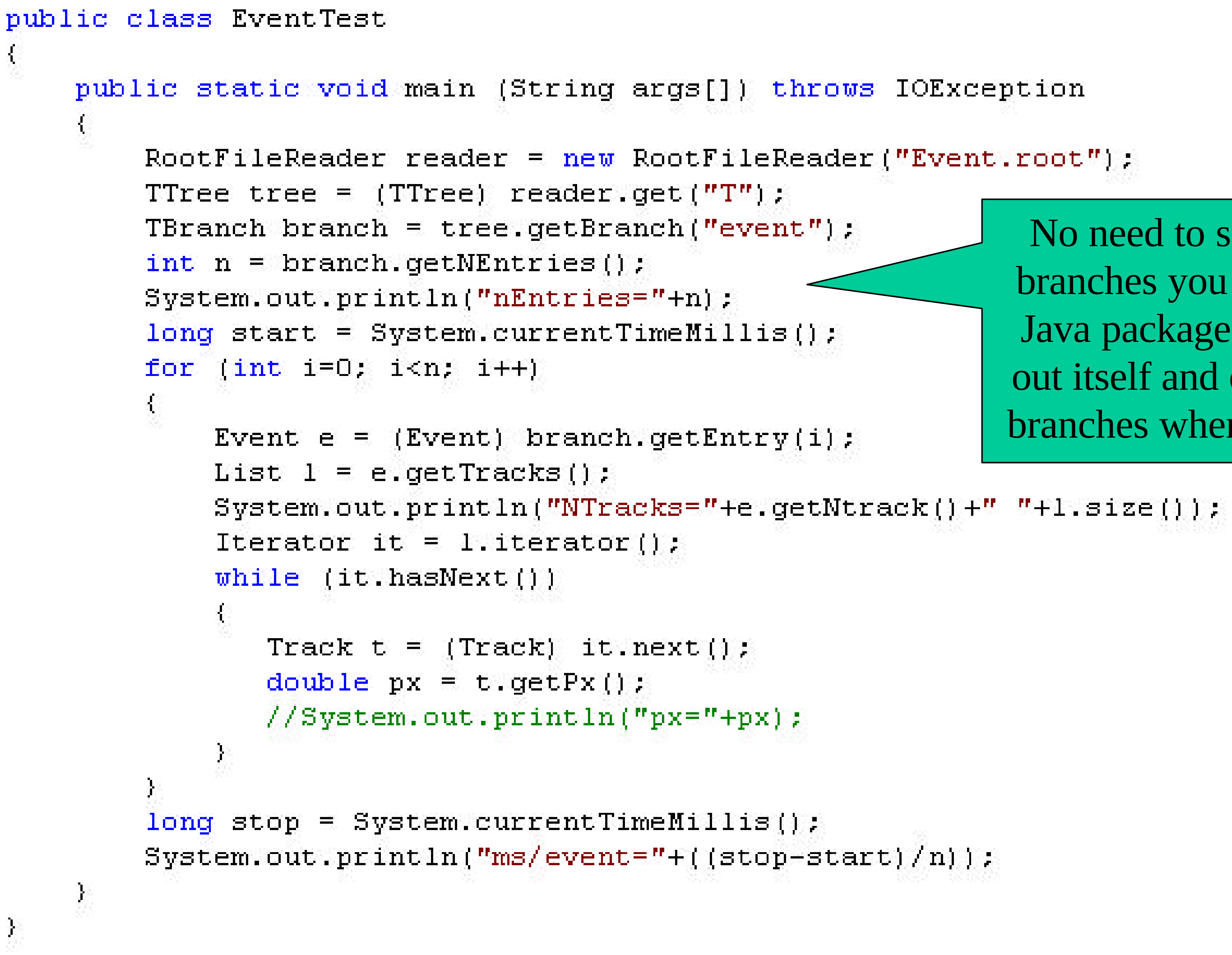
Java Package for Reading Root files

- What?
  - A standalone Java package for *reading* Root files.
  - Gives OO access to *data* in the root file
    - Does not allow calling methods in C++ objects
      - E.g., can create TH1 object, and access TH1 data, but not call TH1.Draw()
  - Works with any Root file written with 3.00/6 or later
  - Needs SteamerInfo data from Root file
    - Works automatically if default Root Streamers used
  - Handles user-defined objects
  - Handles TTree and Splits
- Why?
  - Useful for Java Applets, Servlets, Java GUI's, online systems
  - Can be used with Java based scripting languages (Jython, Beanshell, DynamicJava *etc*).
  - Can be used for Java Event Displays (e.g. Wired)
  - Can be used with Java based analysis tools (e.g. JAS)

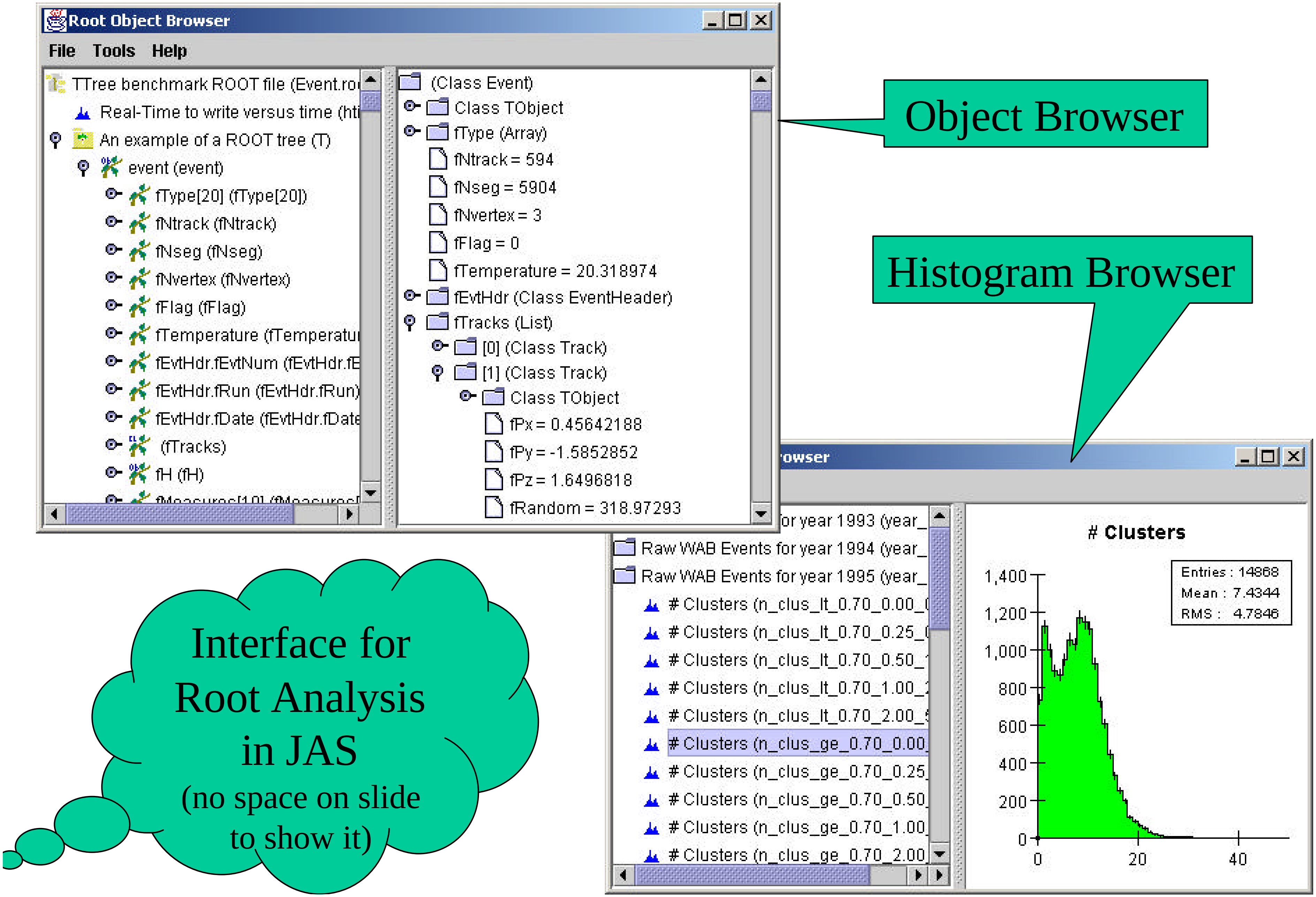
Example Program



```
public class EventTest
{
    public static void main (String args[]) throws IOException
    {
        RootFileReader reader = new RootFileReader("Event.root");
        TTree tree = (TTree) reader.get("T");
        TBranch branch = tree.getBranch("event");
        int n = branch.getNEntries();
        System.out.println("nEntries="+n);
        long start = System.currentTimeMillis();
        for (int i=0; i<n; i++)
        {
            Event e = (Event) branch.getEntry(i);
            List l = e.getTracks();
            System.out.println("NTracks="+e.getNtrack()+" "+l.size());
            Iterator it = l.iterator();
            while (it.hasNext())
            {
               Track t = (Track) it.next();
               double px = t.getPx();
               //System.out.println("px="+px);
            }
        }
        long stop = System.currentTimeMillis();
        System.out.println("ms/event="+((stop-start)/n));
    }
}
```

No need to say which branches you will read. Java package figures it out itself and only reads branches when required.

Demo Applications



Root Object Browser

File Tools Help

TTree benchmark ROOT file (Event.ro
Real-Time to write versus time (ht
An example of a ROOT tree (T)
event (event)
fType[20] (fType[20])
fNtrack (fNtrack)
fNseg (fNseg)
fNvertex (fNvertex)
fFlag (fFlag)
fTemperature (fTemperatu
fEvtHdr.fEvtNum (fEvtHdr.fE
fEvtHdr.fRun (fEvtHdr.fRun)
fEvtHdr.fDate (fEvtHdr.fDate
(fTracks)
fH (fH)
fMeasures[10] (fMeasures[

(Class Event)
Class TObject
fType (Array)
fNtrack = 594
fNseg = 5904
fNvertex = 3
fFlag = 0
fTemperature = 20.318974
fEvtHdr (Class EventHeader)
fTracks (List)
[0] (Class Track)
[1] (Class Track)
Class TObject
fPx = 0.45642188
fPy = -1.5852852
fPz = 1.6496818
fRandom = 318.97293

owser

or year 1993 (year_
Raw WAB Events for year 1994 (year_
Raw WAB Events for year 1995 (year_
Clusters (n_clus_lt_0.70_0.00_
Clusters (n_clus_lt_0.70_0.25_
Clusters (n_clus_lt_0.70_0.50_
Clusters (n_clus_lt_0.70_1.00_
Clusters (n_clus_lt_0.70_2.00_
Clusters (n_clus_ge_0.70_0.00_
Clusters (n_clus_ge_0.70_0.25_
Clusters (n_clus_ge_0.70_0.50_
Clusters (n_clus_ge_0.70_1.00_
Clusters (n_clus_ge_0.70_2.00_

Clusters

Entries : 14868
Mean : 7.4344
RMS : 4.7846

Object Browser

Histogram Browser

Interface for
Root Analysis
in JAS
(no space on slide
to show it)

Status

- Still in Alpha Status
  - Has been tested on as many Root files as we can get our hands on
  - Need more tests (please try it and report bugs)
  - Currently quite slow
    - Easier to test and debug before optimization
    - In future will dynamically create Java (bytecode) objects and use memory mapped IO (new in Java 1.4)
      - Speed should be comparable with C++ Root IO.
- More Info:
  - http://java.freehep.org/lib/freehep/doc/root/index.shtml
  - tonyj@slac.stanford.edu