Doing DST Analyses

Andre Lebedev
MIPP Software Meeting
March 29, 2007

DSTUtil in DSTAnalysis library

- Now that people are doing analyses using DST, it's good to have a standard set of functions/utilities
- DSTUtil is targeted to be incorporated into binaries, but should work in ROOT scripts
  - Binaries will run faster
- dstExample.cc shows how to use those functions to plot number of tracks and vertices

Functions in DSTUtil

- GetFileList()
  - Takes command-line arguments and creates a vector of strings with ROOT file names
  - Arguments can be a text file with ROOT files or ROOT file names
    - You will need file lists if the number of files is too large
- MIPPRunSummary* GetRunSummary(TFile*)
  - Returns the first object from the run summary tree
  - User is responsible for deleting the object!
    - Done intentionally: get the object once from the file and save it

Functions in DSTUtil (cont.)

- TTree* GetEvtTree(TFile*)
  - Get event tree. Finds the tree regardless of the folder name: the first folder that matches FillEventSummary is used
    - Pass 3 tree lives in FillEventSummary_pass3 folder
- TTree* GetSpillTree(TFile*)
  - Get spill tree in the same way as event tree
- SigHandle()
  - Called by GetFileList to set gStop variable to 1

Brief Example

```
int main(int argc, char** argv) {
  vector<string> list;
  GetFileList(argc, argv, list); // Extract file names from the command-line arguments

  TTree* evtTree;
  MIPPEventSummary* evt = new MIPPEventSummary;

  for (unsigned int ifi = 0; ifi < list.size() && !gStop; ++ifi) {
    TFile* f = TFile::Open(list[ifi].c_str());
    if (!f) continue;

    TTree* evtTree = GetEvtTree(f); // Get event tree from the file
    if (!evtTree) continue;

    int nEnt = evtTree->GetEntries();
    evtTree->SetBranchAddress("fvtxconfit.", &evt); // Set address to ConFit branch
    for (int ent = 0; ent < nEnt && !gStop; ++ent) { // gStop will break on Ctrl^C
      evtTree->GetEntry(ent);
  ...
    }
  }
}
```

What's next

- Ideally DSTUtil (or a set of utilities) should define standard cuts that will be used
  - I added HavePileUp120() function to DSTUtil.h
  - Others are welcome to contribute
- `DSTAnalysis/{Validator/PiKRatio}` are already using DSTUtil
- If you think a function or class would be of general use to DST analyses, add it to DSTAnalysis library