

# Simple PAW-Like Histogram/Cut Manager

V. Choutko

CERN, AMS Analysis Meeting April 26,2006

- Motivation
- Method
- Main routines
- Very Simple Cut Manager
- Example of Interactive Histos Reading

**Wanted To Have:**

- PAW-like id (int) based histogram manipulation;
- Easy way to simultaneously read same histos from different root files;
- Preserve access to all root features;
- Easy way to write analysis code;
- Light weighted package;

Separate STL Maps for TH1F, TH2F, TProfile

```
typedef map<AMSID,TH1F*> hb1_d;
typedef map<AMSID,TH1F*>::iterator hb1i;
typedef map<AMSID,TH2F*> hb2_d;
typedef map<AMSID,TH2F*>::iterator hb2i;
typedef map<AMSID,TProfile*> hbp_d;
typedef map<AMSID,TProfile*>::iterator hbpi;
static hb1_d hb1;
static hb2_d hb2;
static hbp_d hbp;
static char CurrentDir[1024];

class AMSID

protected:
    integer _id;
    char * _name;
    ...
```

Booking:<sup>1</sup>

```
AMSEventR::hbook1(int id,const char title[], int ncha, float a, float b);
```

Manipulating:

```
void AMSEventR::hprint(int id, char opt []="");  
void AMSEventR::hlist(char ptit []="");  
void AMSEventR::hreset(int id);  
void AMSEventR::hdelete(int id);  
void AMSEventR::hcopy(char dir [],int id1, int id2);  
void AMSEventR::hdivide(int id1,int id2,int id3);  
void AMSEventR::hsub(int id1,int id2,int id3);  
void AMSEventR::hfit1(int id,char func []);  
void AMSEventR::hadd(int id1,int id2,int id3);  
void AMSEventR::hcdire(const char dir []="");
```

Accessing:

---

<sup>1</sup>Only TH1F\* routines are shown others are similar

```
TH1F* AMSEventR::h1(int id);
```

Reading:

```
void AMSEventR::hfetch(const char file[]);
```

Filling:

```
void AMSEventR::hfill(int id, float a, float b, float w);
```

```
void AMSEventR::hf1(int id, float a, float w=1);
```

Book set of same histos:

```
void AMSEventR::hbook1s(int id,const char title[], int ncha, float a, float b,  
int howmany=6,int shift=100000);
```

Fill Set of Histos

```
bool cuts[ncuts];  
void AMSEventR::hf1s(int id,float a, bool cuts[], int ncuts,int icut,  
int shift=100000,float w=1);  
// Fill the series of histos  
// 1st before cuts  
// before cut icut-1  
// before icut-1 as last cut  
// after icut-1 as last cut  
// after last cut as first cut  
// after cut icut-1 as first cut
```

```
root [0] .x 1.C
root [1] AMSEventR::hfetch("e0510_4.root");
330 1D Histos fetched
11 2D Histos fetched
root [3] AMSEventR::hprint(4);
root [4] AMSEventR::hprint(200004);
root [13] AMSEventR::hfetch("ad_1.root");
332 1D Histos fetched
11 2D Histos fetched
root [14] AMSEventR::hprint(4);
root [15] AMSEventR::hprint(200004);
root [16] AMSEventR::hcdire("e0510_4.root");
root [17] AMSEventR::hprint(200004,"same");
root [26] TH1F *h=AMSEventR::h1(100004);
root [27] h->Draw();
```