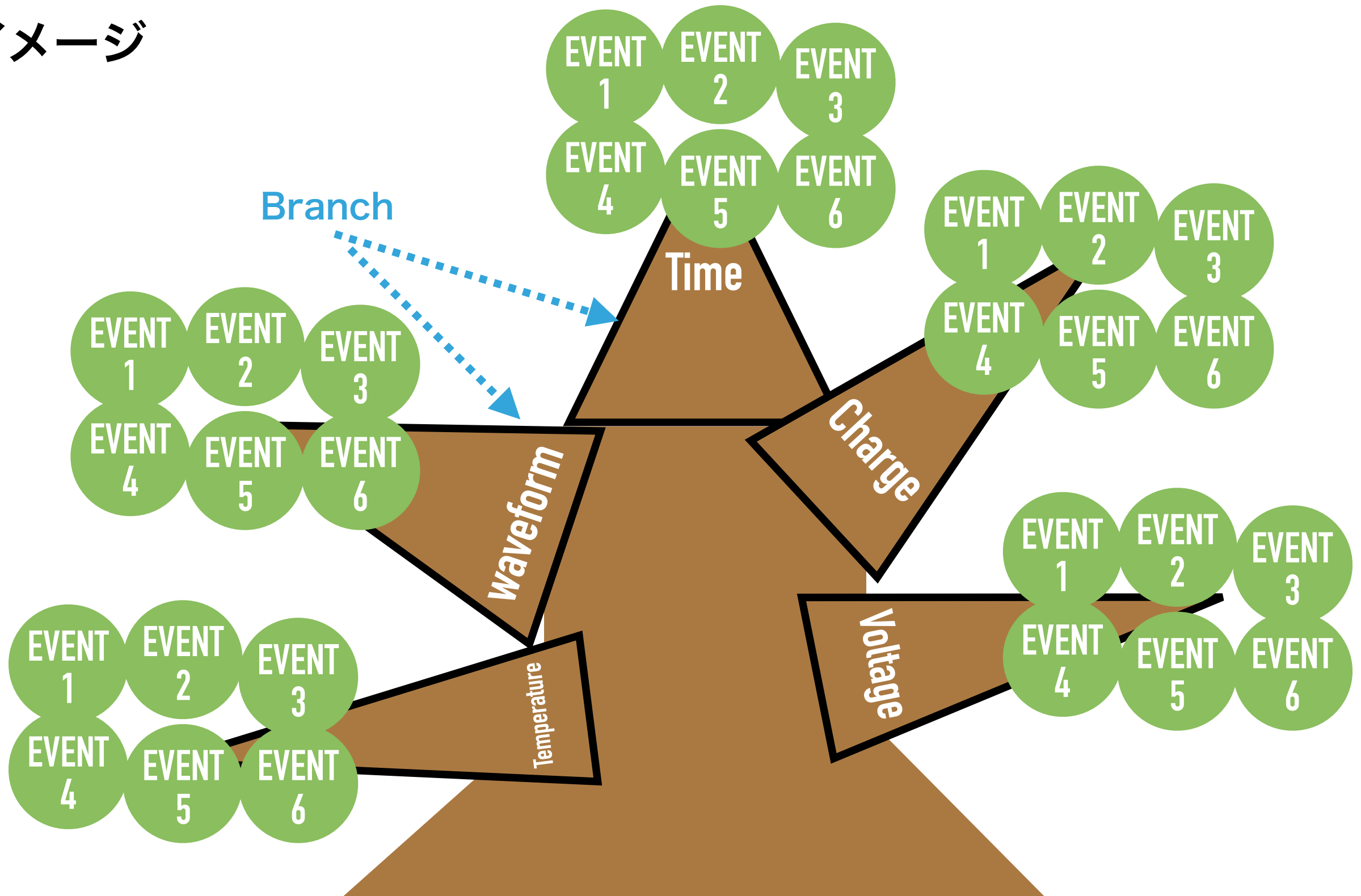


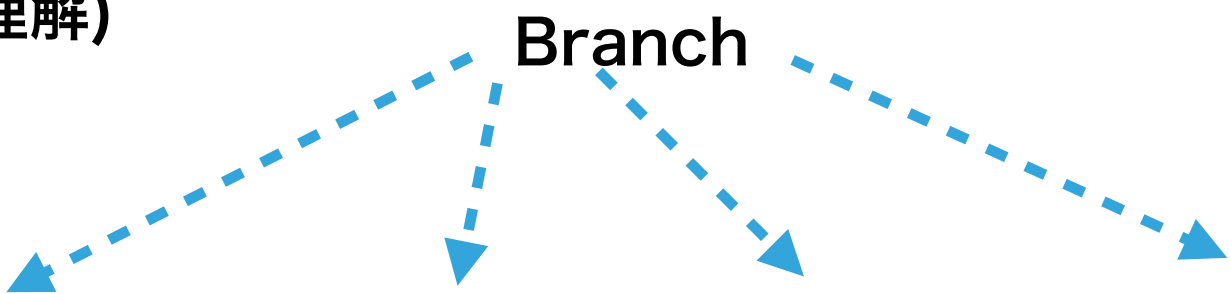
# TTreeの練習

---

- ▶ ROOTの機能
- ▶ イベント毎に情報を整理できる
- ▶ イメージ



- ▶ 表にするとこんな感じ(尾崎の理解)



	waveform	Voltage	Temperature	Charge
event 1	array[1024]	90.5	-100	200
event 2	array[1024]	81.3	-100	400
event 3	array[1024]	72.4	-100	415
...	...	...	...	...
event N	array[1024]	56.8	-100	572

# 中身の簡単な説明

- ▶ shnet1にある「/home/grp1/ozaki.kosuke/228K6.0V.root」を使う。
- ▶ \$ scp (USERNAME)@shnet1.stelab.nagoya-u.ac.jp:/home/grp1/ozaki.kosuke/228K6.0V.root .

## 中身の見方

```
> root 228K6.0V.root

Applying MyStyle settings...

root [0]
Attaching file 228K6.0V.root as _file0...
(TFile *) 0x7f8fb524ad00
root [1] .ls
TFile**      228K6.0V.root
TFile*        228K6.0V.root
KEY: TTree    tree;1 tree
root [2] tree->Print()
*****
*Tree      :tree      : tree
*Entries   : 400000 : Total = 3293616234 bytes File Size = 3291574287 *
*          :          : Tree compression factor = 1.00
*****
*Br   0 :ADC      : ADC[2][1024]/F
*Entries :400000 : Total Size= 3288800375 bytes File Size = 3286133380 *
*Baskets : 133334 : Basket Size= 32000 bytes Compression= 1.00
*.....*
*Br   1 :Temperature : Temperature/I
*Entries : 400000 : Total Size= 1605333 bytes File Size = 1603978 *
*Baskets : 51 : Basket Size= 32000 bytes Compression= 1.00
*.....*
*Br   2 :over_voltege : over_voltage/F
*Entries : 400000 : Total Size= 1605382 bytes File Size = 1604029 *
*Baskets : 51 : Basket Size= 32000 bytes Compression= 1.00
*.....*
*Br   3 :V          : V/F
*Entries : 400000 : Total Size= 1604777 bytes File Size = 1603468 *
*Baskets : 51 : Basket Size= 32000 bytes Compression= 1.00
*****
```

- .ls : でtreeの名前？が見える。  
今回は[tree]
- tree->Print()でevent数や  
Branchが見える
- Entriesがevent数 今回は  
400000event

このデータは温度228K、印加電圧  
100.468Vに固定して測定した  
データなので全てのイベントで  
Temperature = 228 [K]  
V<sub>over</sub>=6 [V]  
V<sub>bias</sub>=100.468 [V]

ADC[2][1024]の中身は波形2ch分  
で中身はイベント毎に違う  
単位：[ns],[mV]

```
1 void tutorial1(){
2   TString filename = "228K6.0V.root"; //ファイル名
3   TFile *f = new TFile(filename, "OPEN"); //228K6.0V.rootというファイルを開く
4   TTree *tree = (TTree*)f->Get("tree"); //treeというTTreeを取り出す
5
6   Float_t waveforms[2][1024]; //TTreeのBranchの中身を入れる変数を用意
7   Int_t T; //変数名はBranch名と同じでもいいし違ってても良い
8   Float_t overV; //型に注意、tree内のover_voltageを入れるつもりなのでFloat_tを用意した。
9   Float_t biasV; //Branchの方はtree->Print()で確認できる。
10
11  tree->SetBranchAddress("ADC", waveforms); //変数とBranchを紐づける。
12  tree->SetBranchAddress("Temperature", &T); //変数のポインタをわたすこと。
13  tree->SetBranchAddress("over_voltege", &overV); //over_voltageでなくてover_voltegeになっているのはroot作ったときのミス
14  tree->SetBranchAddress("V", &biasV);
15
16  //この段階で変数とBranchの紐づけをした。しかし変数になにか値がはいっているわけではない。イベントを指定することで各変数に各Branchの値
17  //イベントの指定: tree->GetEntry(i);
18  //でi番目のイベントの値が各変数に入る。
19
20  tree->GetEntry(332); //event332の値が各変数に入った。
21  //event332の波形をプロットしてみる。
22  Float_t time[1024];
23  for(Int_t i=0; i<1024; i++) time[i]=i;
24  TGraph *g = new TGraph(1024, time, waveforms[0]);
25  g->Draw("AL");
26
27  //tree->GetEntries(); で全イベント数を取得できる。
28  //Int_t nevent;
29  //nevent = tree->GetEntries();
30  //for(Int_t i=0; i<nevent; i++){
31  //tree->GetEntry(i);
32  //で各eventの中身が順番に全て触れる。
33  //}
34 }
```

実行すると波形が出てくるはず

```
1
2
3 //hist.txtの中身をTTreeに詰めるスクリプト
4 void tutorial2(){
5
6     Float_t charge;//tree詰めたい変数定義
7
8     TTree *tree = new TTree("tree","tree");//treeとうTTreeを用意
9     tree->Branch("charge",&charge,"charge/F");//BranchでTTreeにBranchを追加できる.ここでchargeとtreeが紐付けされている。
10
11     ifstream fin("hist.txt");//"hist.txt"を開く
12     while(fin >> charge){//"hist.txt"の中身を1行ずつchargeにいれる。whileは繰り返して、hist.txtの最後の行まで
13         tree->Fill();//これでBranchに使用している変数(charge)の「現在の」値がtreeに詰められる。
14     }
15
16     TString name = "hist.root";//新しくできるrootファイルの名前
17     TFile *fout = new TFile(name,"recreate");
18     tree->Write();//foutというrootファイルにtreeを書き込む
19     fout->Close();//閉じる
20
21 }
```

実行するとhist.rootができるはず

hist.txtも220K6.0V.rootと同じ場所に置いておく