

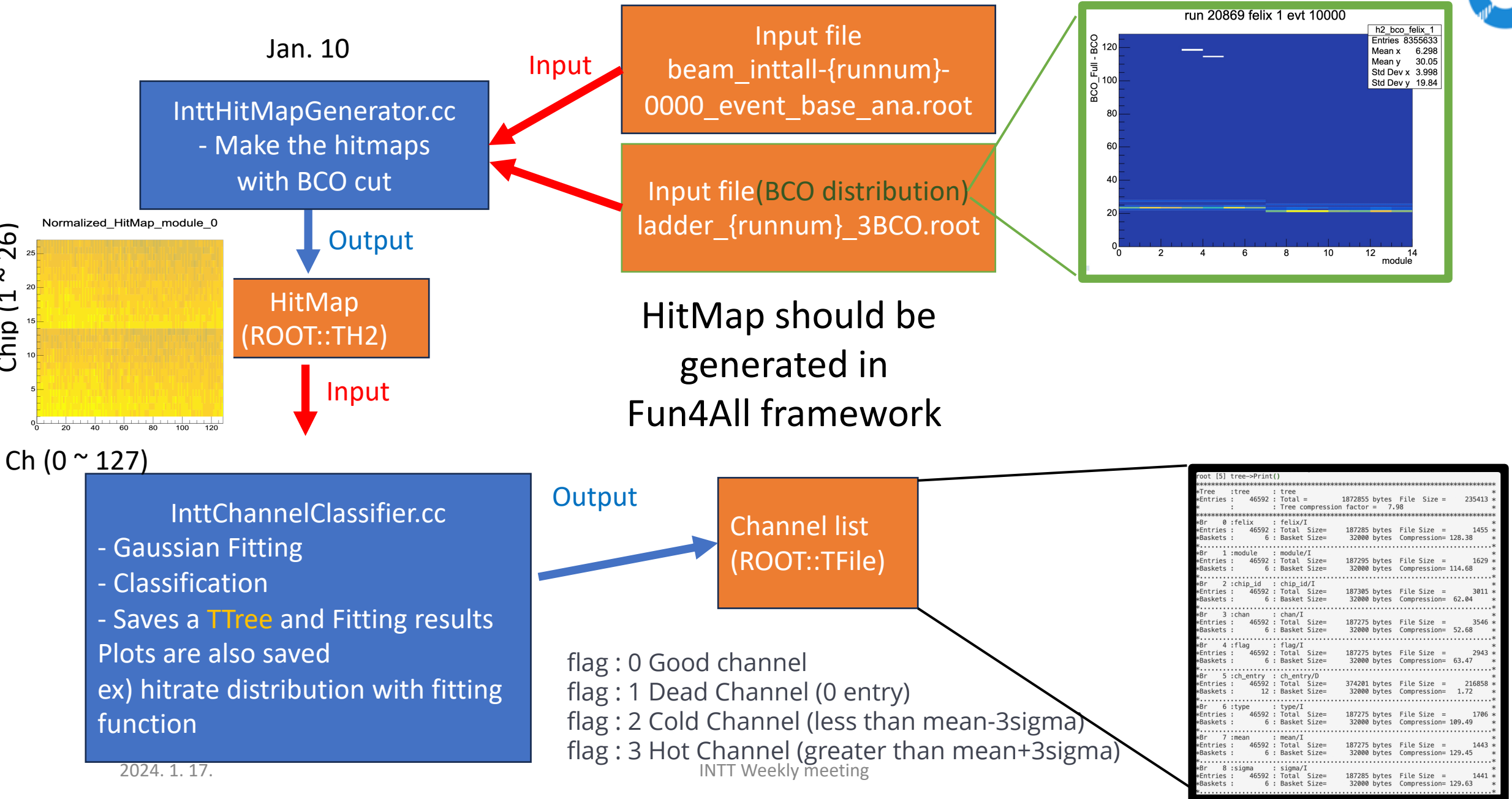
Update on Fun4All based Channel Classifier

Jaein Hwang
Byungsik Hong

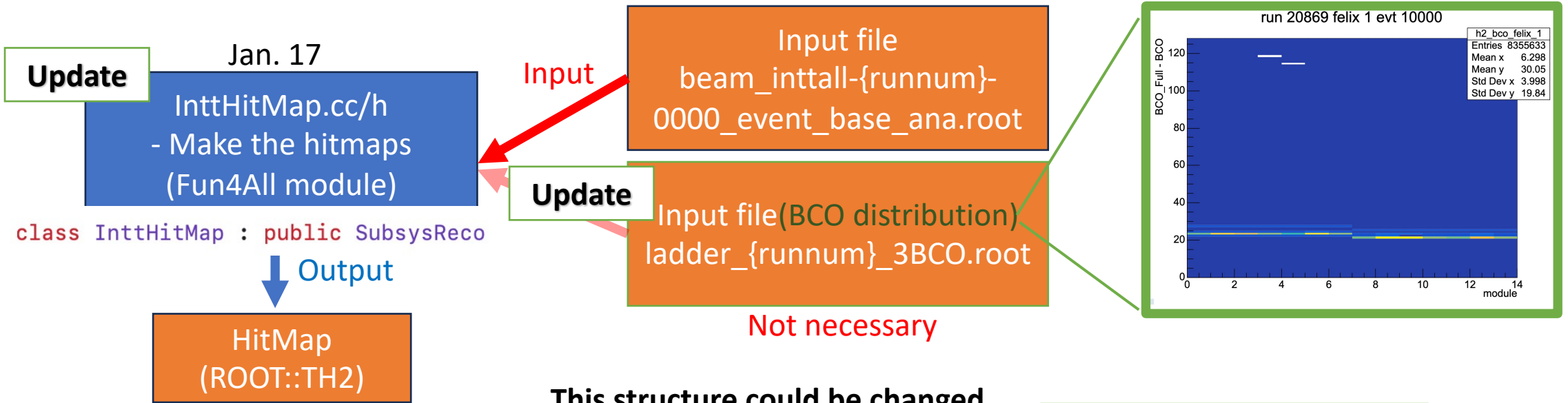


https://github.com/gwd213/INTT/tree/main/general_codes/Jaein/Fun4all/GaussianClassifier

Structure of Hot channel classifier code(Old Ver.)

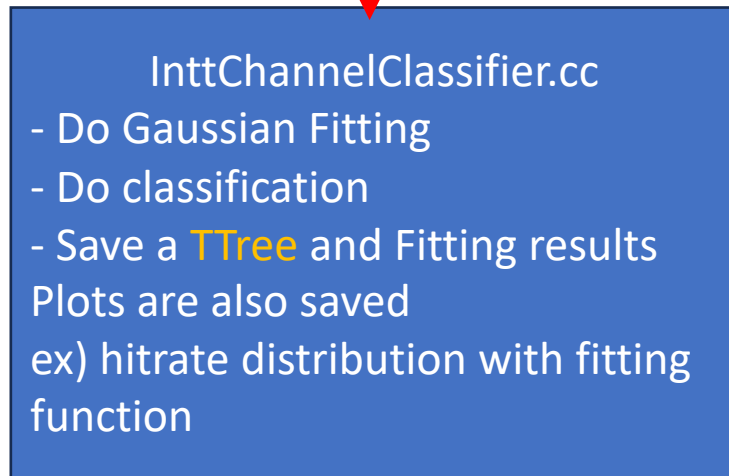


Structure of Hot channel classifier code(Jan. 17)



This structure could be changed.
(Possibly I missed something.)

Development is needed



Output

Channel list
(ROOT::TFile)

Input

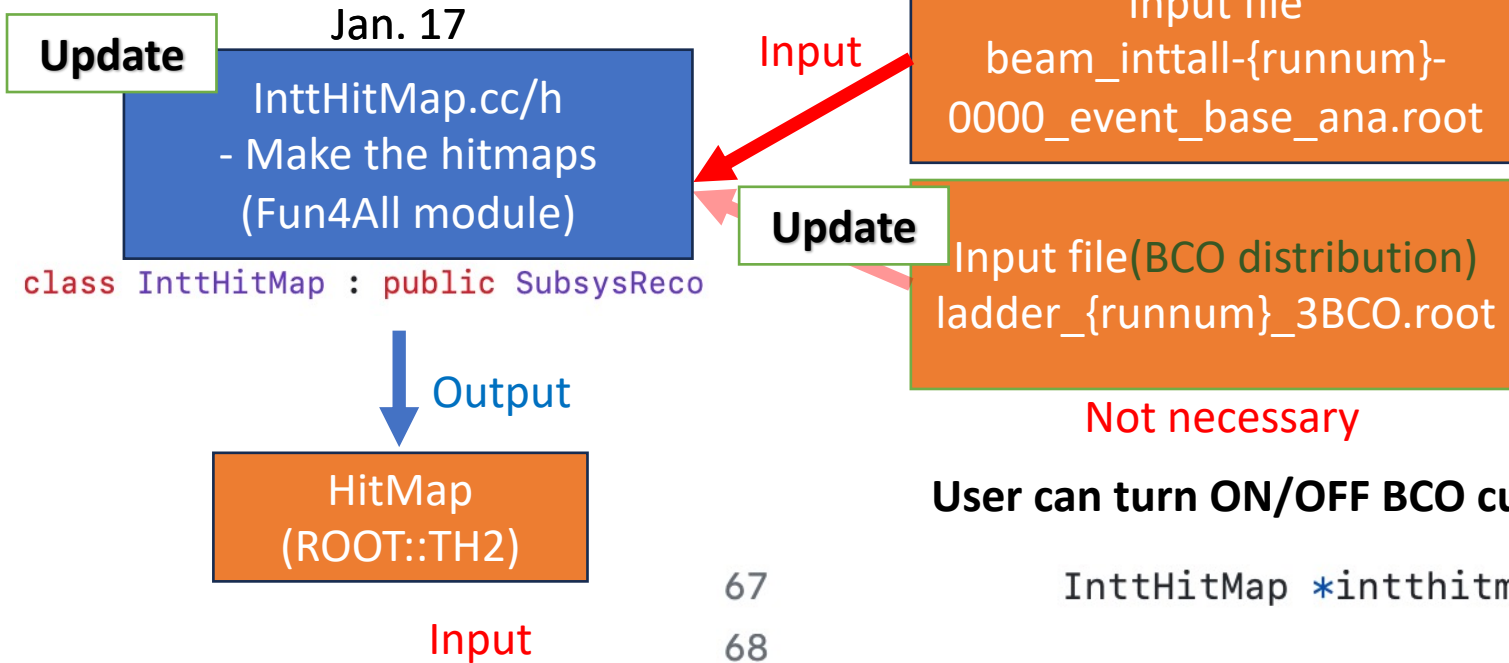
InttHotDeadMap.cc/h

C++ Class
available in Fun4All
(like InttHotMap.cc/h)

flag : 0 Good channel
flag : 1 Dead Channel (0 entry)
flag : 2 Cold Channel (less than mean-3sigma)
flag : 3 Hot Channel (greater than mean+3sigma)

https://github.com/SPHENIX-Collaboration/INTT/tree/main/general_codes/hachiya/F4AInttRead/src/InttHotMap.h

InttHitMap.cc/h



User can turn ON/OFF BCO cut option in Fun4All macro.

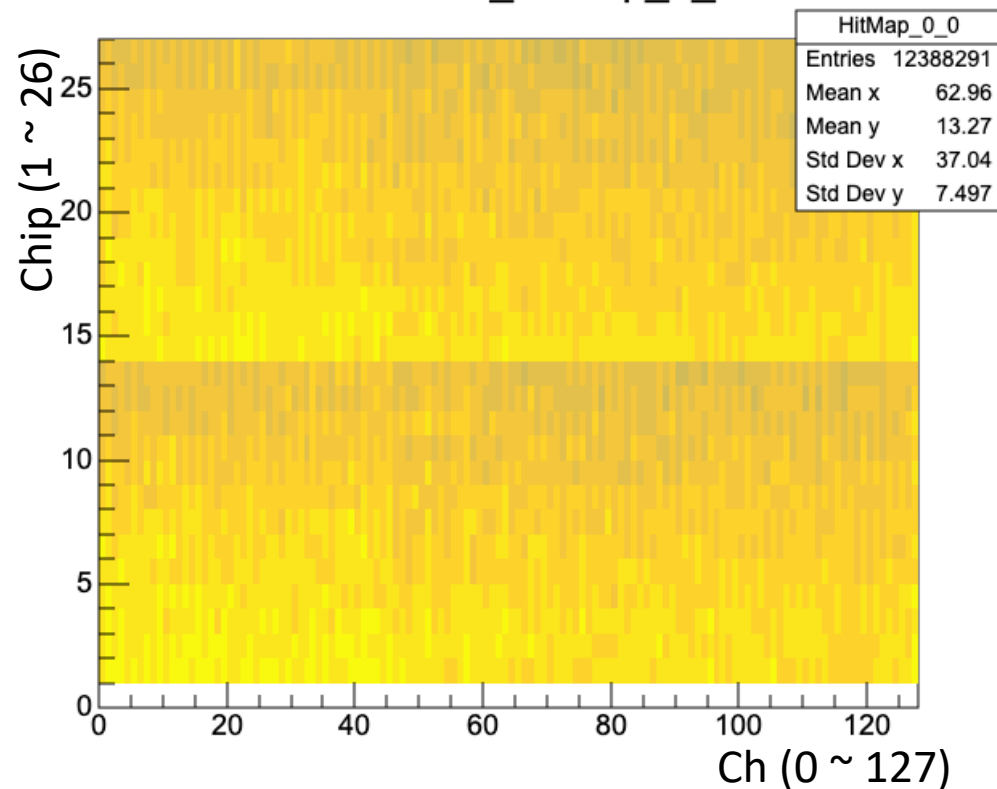
```
67      InttHitMap *intthitmap = new InttHitMap("intthitmap",
68                                              i_file.c_str(),
69                                              o_file.c_str(),
70                                              nevents);
71
72      intthitmap->SetBCOCut(bcoflag);
73      intthitmap->SetBCOfile(bco_file.c_str());
74      se->registerSubsystem(intthitmap);
```

True or false

Confirmation of HitMap after moving to Fun4All

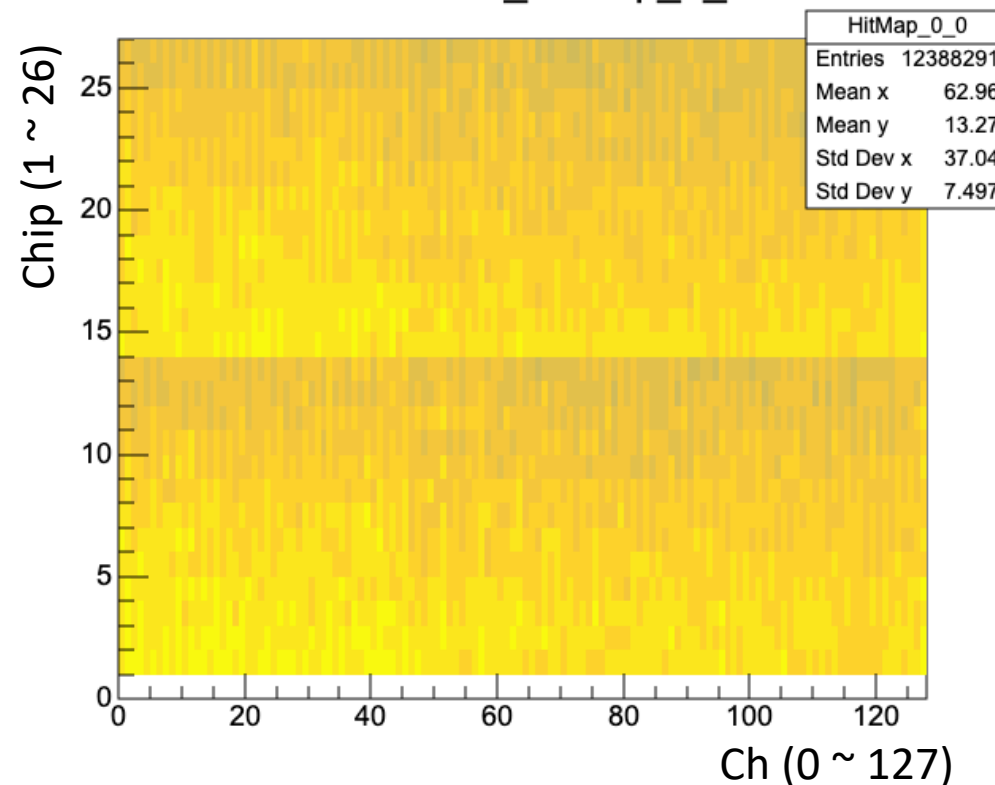
HitMap generated by
old version code(out of fun4all framework)

Normalized_HitMap_0_0



HitMap generated by
new version module(fun4all framework)

Normalized_HitMap_0_0



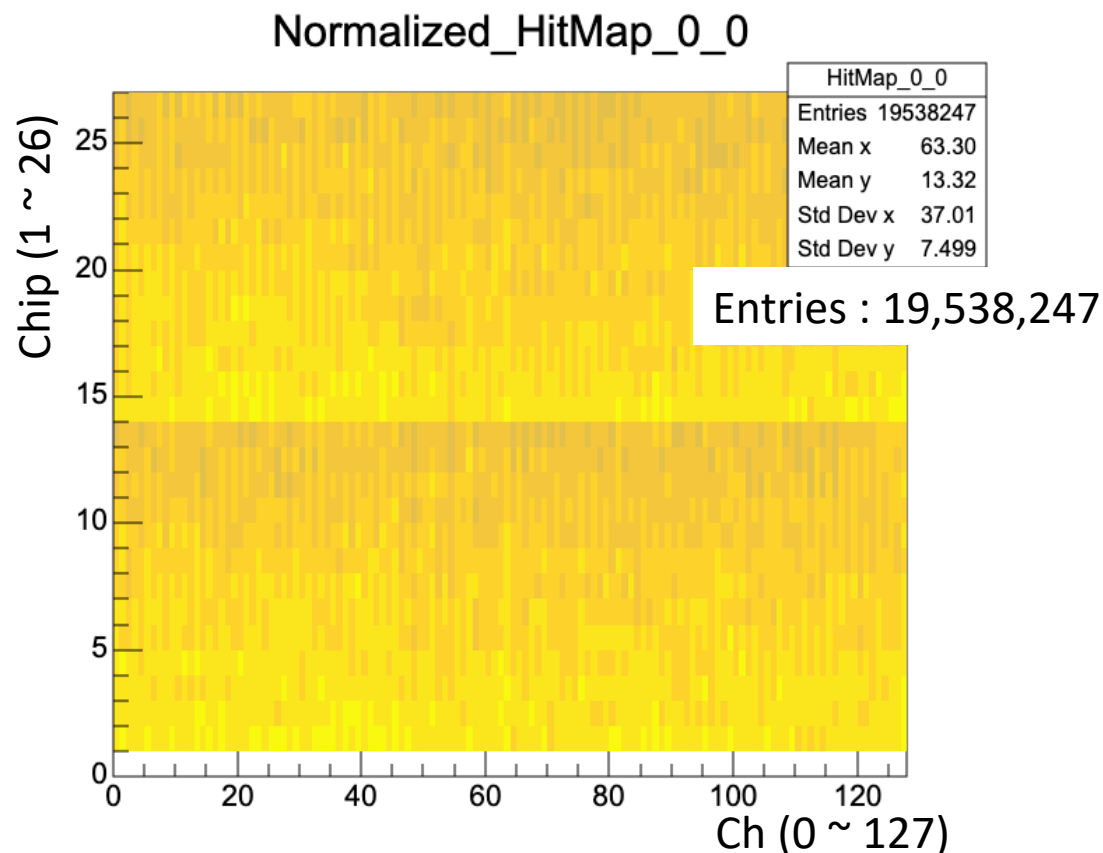
The results before and after appear the same as each other.

Migration of framework will not cause the difference between before and after.

Confirmation of HitMap after moving to Fun4All

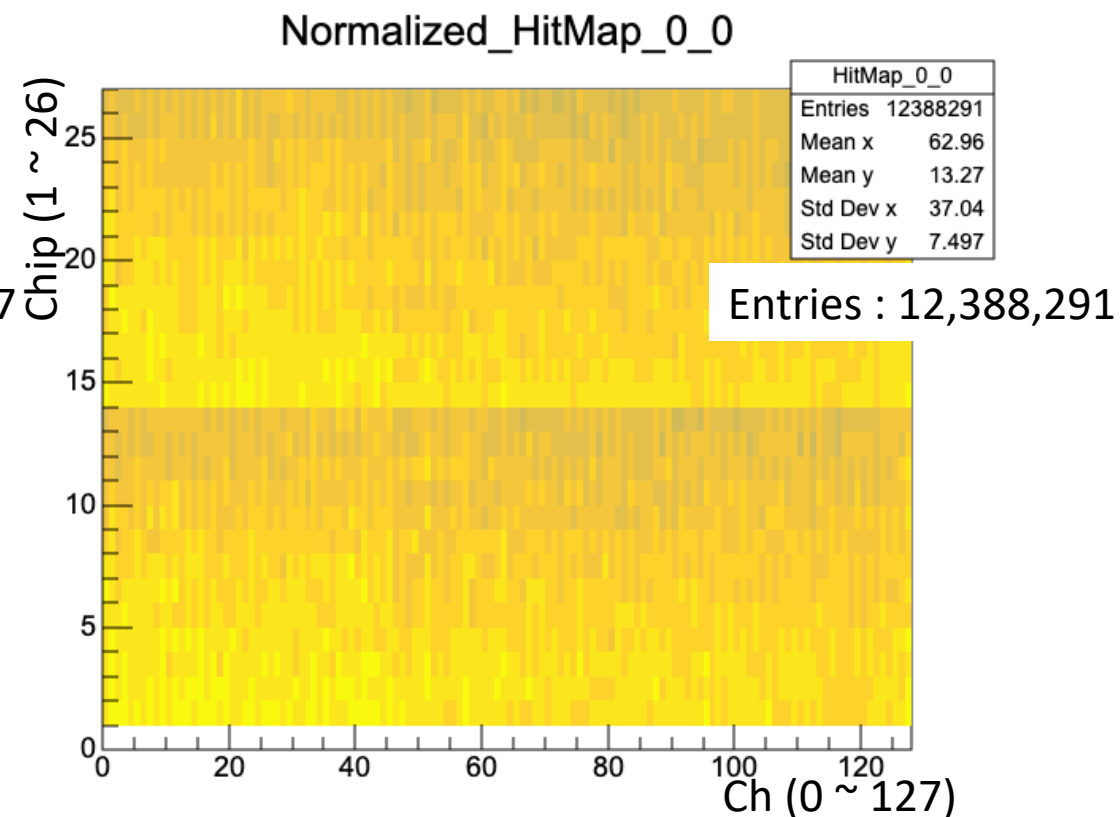
New version(fun4all framework)

without BCO cut



New version(fun4all framework)

with BCO cut



of entries without BCO cut is much higher than the one with BCO cut.
The Fun4All module with BCO cut option seems to be working properly.

Summary and Plan



- 1st version of FUN4All based HitMap creator has published.

https://github.com/gwd213/INTT/tree/main/general_codes/Jaein/Fun4all/GaussianClassifier

README and more comments in the code will be added.

Confirmation of the **way to generate a HitMap & structure of input file** is needed.

Name	Last commit message	Last commit date
 ..		
 macro	First commit for Gaussian Classifier	2 days ago
 src	*Dict* were replaced to InttEventLinkDef.h	yesterday

Plan

- Development of HotDeadMap.cc/h (C++ Class available in Fun4All) to load the hot/dead/cold channels

By using this class, user will be able to choose which sigma cut($\geq 3\sigma$) would be used in their analysis.

backup

Hot Channel algorithm

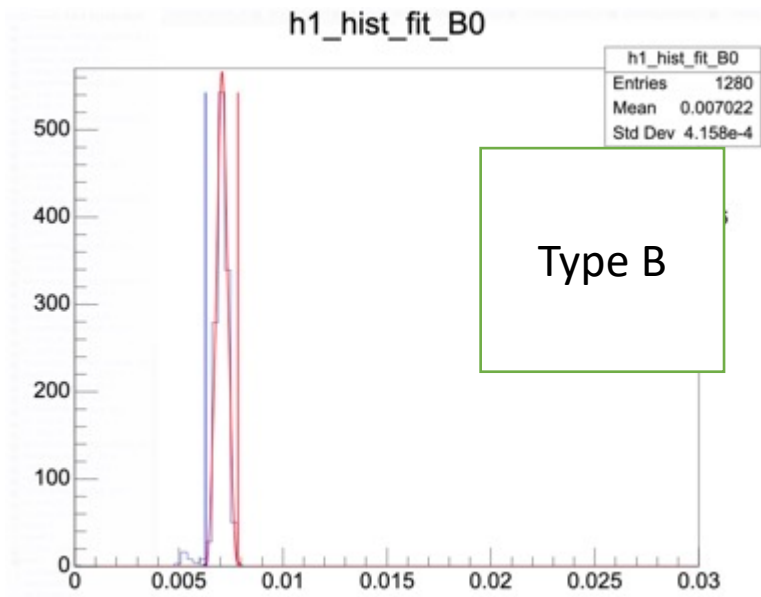
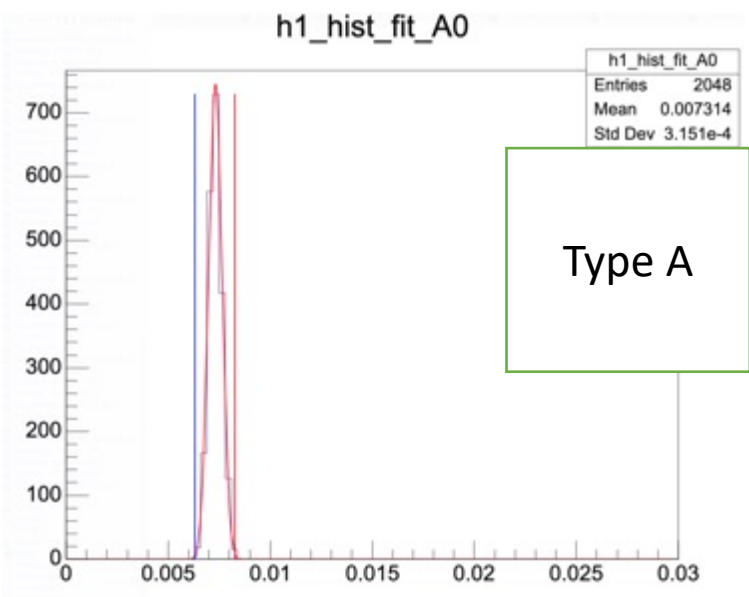
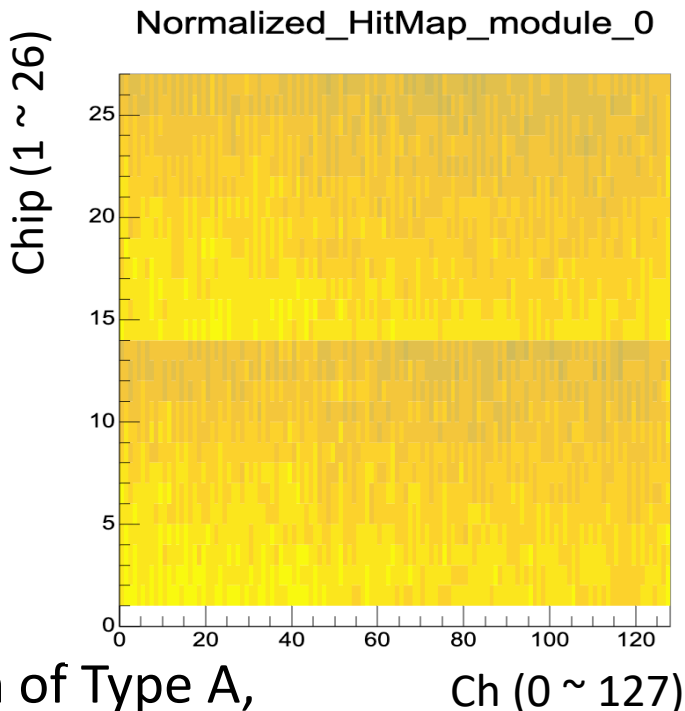
Reminder : Procedure of hot channel classifier

1. Make a normalized hit map distribution
normalized by :
 - number of event
 - Acceptance difference depending on the chip type (Type A and Type B) and layer (inner and outer)

2. Draw Hit rate distribution for every half ladders.
Each half ladder have two hit rate distributions, one is distribution of Type A,
the other is for Type B.

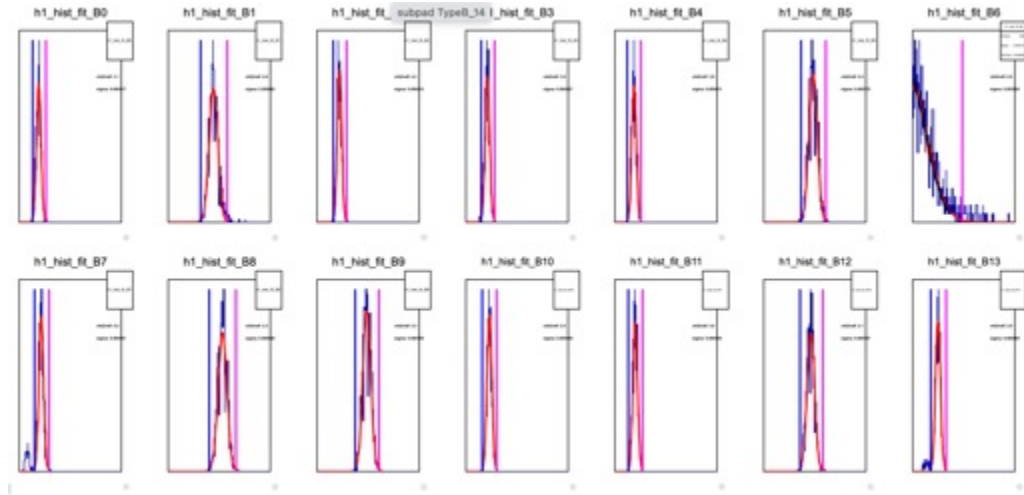
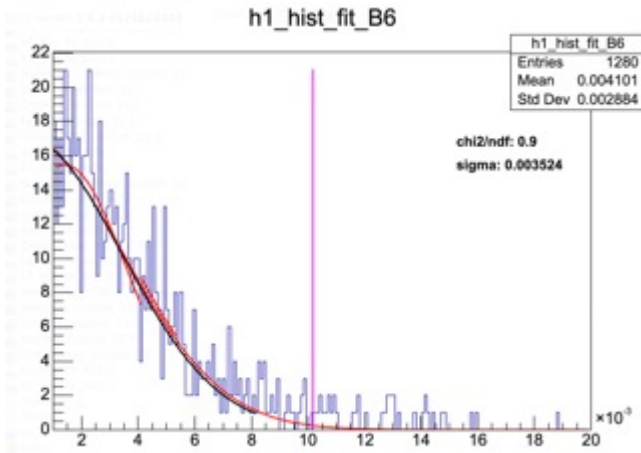
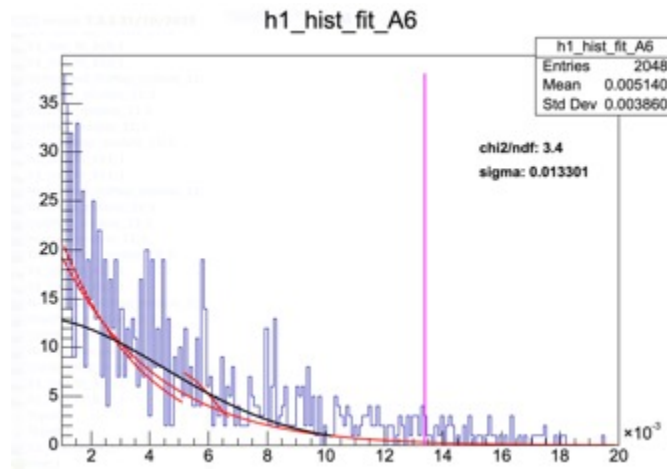
3. Definition of channel

- Hot Channel : mean + 3sigma
- Cold Channel : mean – 3sigma
- Dead Channel : 0 hit



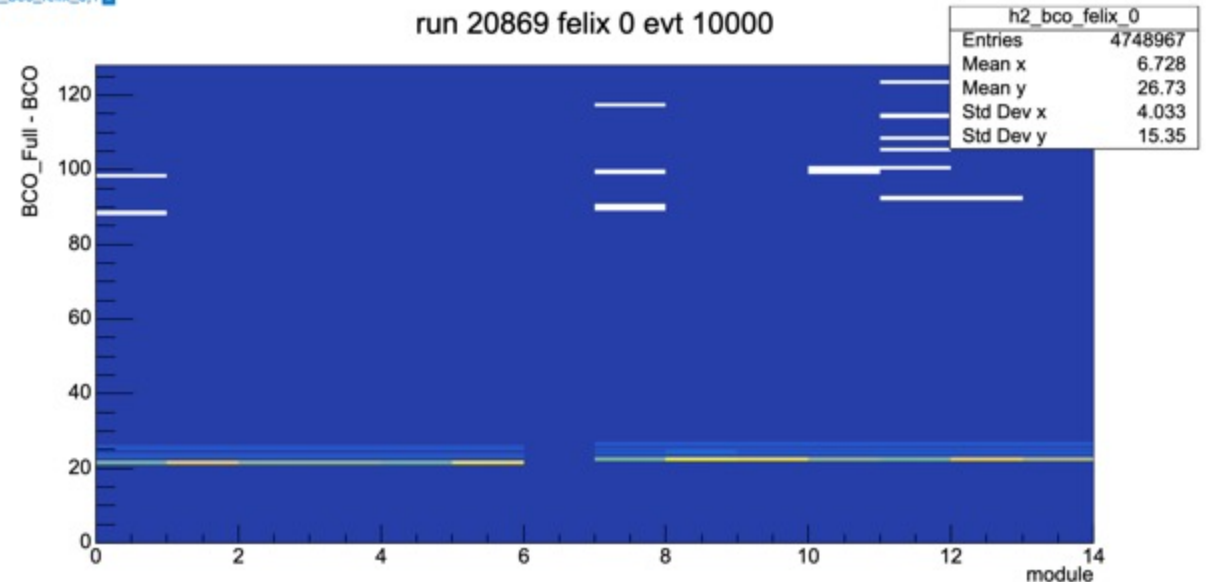
Problematic ladders(Felix 0 module 6)

Hit rate distribution before BCO cut – No clear time peak in BCO distribution



h2_bco_felix_0

run 20869 felix 0 evt 10000



Effect of BCO cut

Hit rate distribution After/Before BCO cut

