

ハイレベルフィルター実装記録

2024/7/12

7/11のオーナーセミナーで収集したデータをもとにDataReplayしてみる。
実装予定のフィルターは以下の通りである。

- TOT
- Multiplicity
- TOF
- Tracking

更に詳細に示す。

デバック項目

- Map
 - loadingwiremap関数
 - findwrield関数
- Multiplicity
 - 任意のクラスター数(コンパイルなしで変更できるように)
 - 任意のクラスター要素数(コンパイルなしで変更できるように)
- TOFトリガー信号(pla1を想定)
- Tracking(TMWDCTrackingProessorを参考にこれは1トラックのみ)
 - Multiplicityフィルターの後段に接続

チェック項目

- フィルターの処理時間
- 本物を削らないかどうか？
 - ミニマム要件では電荷フィルター
 - E585の6Liゲートをもとに、VDCのヒットパターン分布を表示、multiplicityフィルターのゲート条件に反映させる。
-

昨日のオーナーセミナーで動かした時、なぜフィルター後のデータが0にならなかった。
まず今のコードが正しく動くか確かめてみる。(実験室のパソコンではデバック文がバグって文字化けしていたため、原因がわからなかった)

1: DataReplayer で正しくリプレイする。

TFBFilePlayer -> High Filter -> Sink

修正完了

```
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] daq::service::MetricsPlugin SendSocketMetrics done
```

```
[[01;36m14:22:47[0m]][[01;32mINFO[0m] in[0]: in: 1000 (0.648 MB) out: 0 (0 MB)
```

```
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] daq::service::MetricsPlugin SendSocketMetrics done
```

```
[[01;36m14:22:47[0m]][[01;32mINFO[0m] out[0]: in: 0 (0 MB) out: 500 (0.396 MB)
```

```
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] Exception caught in ProcessWork(), going to Error state
```

```
[[01;36m14:22:47[0m]][[01;32mINFO[0m] daq::service::Plugin state : ERROR
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] daq::service::Plugin run number (from redis) = 119
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] daq::service::MetricsPlugin state change: ERROR
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] Shutting down Plugin Manager
[[01;36m14:22:47[0m]][[01;34mDEBUG[0m] daq::service::ParameterConfigPlugin bye
```

セカンドフィルター内に何か問題がある。

60行目に問題あり

SubTimeFrameコンテナには何もデータが入ってない。

```
auto& hbf = SubTimeFrame->at(0);
std::cout << "point1-3" << std::endl;
```

SubTimeFrameヘッダーを確認する。

最悪、手で生データのバイナリ解析をする必要がある。

まずE50の時動いたフィルターを動かしてみる。

やはり動いた。

なぜ？

一回R199.datのデータを手で解析してみる。

JparcE50の時のデータ構造と比較してみる。RUN1088

RUN199

```
0000000 004b4e53454c4946 00000130000000130 >FILESNK.0...0...<
0000020 000000000000000063 0000000000000000c7 >c.....<
0000040 000000000668f81f9 000000000000000000 >...f.....<
0000060 6b6e6953656c6946 682e726564616548 >FileSinkHeader.h<
0000100 0000007473657420 000000000000000000 > test.....<
0000120 000000000000000000 000000000000000000 >.....<
*
0000460 004d5246454d4954 00010018000000388 >TIMEFRM.....<
0000500 0000000a00ef75dc 0049474f4c544c46 >.u.....FLTLOGI.<
0000520 00000038000000050 0000000100ef75dc >P...8....u.....<
0000540 0000000500000001 ff00ff0000000086c >.....l.....<
0000560 00000000668f8209 00061cf341f93835 >...f....58.A....<
0000600 00454d4954475254 0000001000000018 >TRGTIME.....<
0000620 aa0000000001e4c4 00454d4954425553 >.....SUBTIME.<
0000640 00000030000000050 0000000500ef75dc >P...0....u.....<
0000660 00000004c0a802a9 00000000668f8209 >.....f.....<
0000700 0000000000072426 0054414542545248 >&$.....HRTBEAT.<
0000720 0000001000000020 7000000000ef75dc > .....u.....p<
0000740 7800000091800490 00454d4954425553 >.....xSUBTIME.<
```

0000760 0000003000000050 0000000600ef75dc >P...0....u.....<
0001000 00000004c0a802a8 00000000668f8209 >.....f....<
0001020 0000000000074e23 0054414542545248 >#N.....HRTBEAT.<
0001040 0000001000000020 7000000710ef75dc >u.....p<
0001060 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0001100 0000003000000050 0000000600ef75dc >P...0....u.....<
0001120 00000004c0a802a6 00000000668f8209 >.....f....<
0001140 000000000007fb1b 0054414542545248 >.....HRTBEAT.<
0001160 0000001000000020 70000007b0ef75dc >u.....p<
0001200 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0001220 0000003000000050 0000000600ef75dc >P...0....u.....<
0001240 00000004c0a802a7 00000000668f8209 >.....f....<
0001260 0000000000096b8f 0054414542545248 >.k.....HRTBEAT.<
0001300 0000001000000020 7000000828ef75dc >u.(...p<
0001320 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0001340 0000003000000050 0000000600ef75dc >P...0....u.....<
0001360 00000004c0a802a4 00000000668f8209 >.....f....<
0001400 00000000000a556f 0054414542545248 >oU.....HRTBEAT.<
0001420 0000001000000020 70000007b0ef75dc >u.....p<
0001440 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0001460 0000003000000050 0000000600ef75dc >P...0....u.....<
0001500 00000004c0a802a2 00000000668f8209 >.....f....<
0001520 00000000000a8cb4 0054414542545248 >.....HRTBEAT.<
0001540 0000001000000020 7000000828ef75dc >u.(...p<
0001560 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0001600 0000003000000050 0000000600ef75dc >P...0....u.....<
0001620 00000004c0a802a1 00000000668f8209 >.....f....<
0001640 00000000000aa96a 0054414542545248 >j.....HRTBEAT.<
0001660 0000001000000020 70000007b0ef75dc >u.....p<
0001700 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0001720 0000003000000050 0000000600ef75dc >P...0....u.....<
0001740 00000004c0a802a3 00000000668f8209 >.....f....<
0001760 00000000000aada3 0054414542545248 >.....HRTBEAT.<
0002000 0000001000000020 70000007b0ef75dc >u.....p<
0002020 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0002040 0000003000000050 0000000600ef75dc >P...0....u.....<
0002060 00000004c0a802a5 00000000668f8209 >.....f....<
0002100 00000000000abda5 0054414542545248 >.....HRTBEAT.<
0002120 0000001000000020 7000000878ef75dc >u.x...p<
0002140 7800000000000000 00454d4954425553 >.....xSUBTIME.<
0002160 0000003000000050 0000000600ef75dc >P...0....u.....<
0002200 00000004c0a802a0 00000000668f8209 >.....f....<
0002220 00000000000b5783 0054414542545248 >.W.....HRTBEAT.<
0002240 0000001000000020 7000000000ef75dc >u.....p<

0002260 7800000000000000 004d5246454d4954 >.....xTIMEFRM.<
0002300 00020018000002a8 0000000a00ef75dc >.....u.....<
0002320 00454d4954425553 00000030000000050 >SUBTIME.P...0...<
0002340 0000000500ef75dc 00000002c0a802a9 >.u.....<
0002360 00000000668f8209 0000000000072426 >...f....&\$.....<
0002400 0054414542545248 0000001000000020 >HRTBEAT.<
0002420 2c582df55e4c7623 2c500f5a9e4c9c5e >#vL^.-X,^.L.Z.P,<
0002440 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0002460 0000000600ef75dc 00000002c0a802a8 >.u.....<
0002500 00000000668f8209 0000000000074e23 >...f....#N.....<
0002520 0054414542545248 0000001000000010 >HRTBEAT.....<
0002540 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0002560 0000000600ef75dc 00000002c0a802a6 >.u.....<
0002600 00000000668f8209 000000000007fb1b >...f.....<
0002620 0054414542545248 0000001000000010 >HRTBEAT.....<
0002640 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0002660 0000000600ef75dc 00000002c0a802a7 >.u.....<
0002700 00000000668f8209 0000000000096b8f >...f....k.....<
0002720 0054414542545248 0000001000000010 >HRTBEAT.....<
0002740 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0002760 0000000600ef75dc 00000002c0a802a4 >.u.....<
0003000 00000000668f8209 00000000000a556f >...f....oU.....<
0003020 0054414542545248 0000001000000010 >HRTBEAT.....<
0003040 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0003060 0000000600ef75dc 00000002c0a802a2 >.u.....<
0003100 00000000668f8209 00000000000a8cb4 >...f.....<
0003120 0054414542545248 0000001000000010 >HRTBEAT.....<
0003140 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0003160 0000000600ef75dc 00000002c0a802a1 >.u.....<
0003200 00000000668f8209 00000000000aa96a >...f....j.....<
0003220 0054414542545248 0000001000000010 >HRTBEAT.....<
0003240 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0003260 0000000600ef75dc 00000002c0a802a3 >.u.....<
0003300 00000000668f8209 00000000000aada3 >...f.....<
0003320 0054414542545248 0000001000000010 >HRTBEAT.....<
0003340 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0003360 0000000600ef75dc 00000002c0a802a5 >.u.....<
0003400 00000000668f8209 00000000000abda5 >...f.....<
0003420 0054414542545248 0000001000000010 >HRTBEAT.....<
0003440 00454d4954425553 00000030000000040 >SUBTIME.@...0...<
0003460 0000000600ef75dc 00000002c0a802a0 >.u.....<
0003500 00000000668f8209 00000000000b5783 >...f....W.....<

0003520 0054414542545248 0000001000000010 >HRTBEAT.....<

0003540 004d5246454d4954 00010018000003a0 >TIMEFRM.....<
0003560 0000000a00ef75dc 0049474f4c544c46 >.u.....FLTLOGI.<
0003600 0000003800000068 0000000400ef75dc >h...8....u.....<
0003620 0000000500000001 ff00ff000000086c >.....l.....<
0003640 0000000668f8209 00061cf341f93835 >...f....58.A....<
0003660 00454d4954475254 0000001000000030 >TRGTIME.0.....<
0003700 aa000000000725c aa00000000011ce7 >\r.....<
0003720 aa00000000015982 aa0000000001ad48 >.Y.....H.....<
0003740 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0003760 0000000500ef75dc 00000004c0a802a9 >.u.....<
0004000 0000000668f8209 0000000000072426 >...f....&\$.....<
0004020 0054414542545248 0000001000000020 >HRTBEAT.<
0004040 7000000000ef75dd 78000000a0000500 >.u....p.....x<
0004060 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0004100 0000000600ef75dc 00000004c0a802a8 >.u.....<
0004120 0000000668f8209 0000000000074e23 >...f....#N.....<
0004140 0054414542545248 0000001000000020 >HRTBEAT.<
0004160 7000000710ef75dd 78000000398001d0 >.u....p...9...x<
0004200 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0004220 0000000600ef75dc 00000004c0a802a6 >.u.....<
0004240 0000000668f8209 000000000007fb1b >...f.....<
0004260 0054414542545248 0000001000000020 >HRTBEAT.<
0004300 70000007b0ef75dd 780000000b000058 >.u....pX.....x<
0004320 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0004340 0000000600ef75dc 00000004c0a802a7 >.u.....<
0004360 0000000668f8209 0000000000096b8f >...f....k.....<
0004400 0054414542545248 0000001000000020 >HRTBEAT.<
0004420 7000000828ef75dd 7800000000000000 >.u.(...p.....x<
0004440 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0004460 0000000600ef75dc 00000004c0a802a4 >.u.....<
0004500 0000000668f8209 00000000000a556f >...f....oU.....<
0004520 0054414542545248 0000001000000020 >HRTBEAT.<
0004540 70000007b0ef75dd 7800000000000000 >.u....p.....x<
0004560 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0004600 0000000600ef75dc 00000004c0a802a2 >.u.....<
0004620 0000000668f8209 00000000000a8cb4 >...f.....<
0004640 0054414542545248 0000001000000020 >HRTBEAT.<
0004660 7000000828ef75dd 7800000000000000 >.u.(...p.....x<
0004700 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0004720 0000000600ef75dc 00000004c0a802a1 >.u.....<
0004740 0000000668f8209 00000000000aa96a >...f....j.....<

```
0004760 0054414542545248 0000001000000020 >HRTBEAT. ....<
0005000 70000007b0ef75dd 7800000000000000 >.u....p.....x<
0005020 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0005040 0000000600ef75dc 00000004c0a802a3 >.u.....<
0005060 00000000668f8209 00000000000aada3 >...f.....<
0005100 0054414542545248 0000001000000020 >HRTBEAT. ....<
0005120 70000007b0ef75dd 7800000000000000 >.u....p.....x<
0005140 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0005160 0000000600ef75dc 00000004c0a802a5 >.u.....<
0005200 00000000668f8209 00000000000abda5 >...f.....<
0005220 0054414542545248 0000001000000020 >HRTBEAT. ....<
0005240 7000000878ef75dd 7800000000000000 >.u.x...p.....x<
0005260 00454d4954425553 0000003000000050 >SUBTIME.P...0...<
0005300 0000000600ef75dc 00000004c0a802a0 >.u.....<
0005320 00000000668f8209 00000000000b5783 >...f.....W.....<
0005340 0054414542545248 0000001000000020 >HRTBEAT. ....<
0005360 7000000000ef75dd 7800000000000000 >.u....p.....x<
0005400 004d5246454d4954 00020018000002a8 >TIMEFRM.....<
```

RUN1088

RUN199で動かすとFrameContainer.hで定義されているSTFコンテナの中のHBF構造体にデータが詰まってないとエラー表示された。

RUN200で動かすと解消された。

2024/7/14

フィルターを機能別に分離する。

現在、FilterTimeFrameSliceABC.cxxにFlt.cxx(ハイレベルフィルター用のクラス)を登録している。
Flt.cxxにはTOTフィルター、TOFフィルター、マルチプリシティ(クラスタ)フィルタを登録している。

これをまず分離する。

トポロジーで各フィルターを接続できるようにする。

→ 完了

0: TO DO

- ☒ LOG(error) << "Failed to queue output channel" 問題
- ☒ フィルターを分離
- ☒ テスト(試しにトポロジーTOT>マルチプリシティ)に変更する。
 - ☐ トポロジーyamlを編集するためのドキュメント整備
- ☒ メモリーリークチェック

トポロジーの変更に伴い、各種スクリプトを変更する必要がある。これが非常にめんどくさい。もうちょっといい方法がないか考えてみる。

トポロジーの変更に伴い変更する必要があるスクリプトファイル

- ☐ mq-param
- ☐ run.sh
- ☐ 親スクリプトの編集(stf-tf-fl.shなど)

LOGICFilter

何もデータが送られてきていない。

```
[14:04:17][STATE] READY ---> RUNNING
[14:04:17][INFO] daq::service::Plugin state : RUNNING
[14:04:17][DEBUG] daq::service::Plugin run number (from redis) = 200
[14:04:17][DEBUG] daq::service::MetricsPlugin state change: RUNNING
[14:04:17][INFO] fair::mq::Device running...
[14:04:18][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
[14:04:18][INFO] in[0]: in: 0 (0 MB) out: 0 (0 MB)
[14:04:18][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
[14:04:18][INFO] out[0]: in: 0 (0 MB) out: 0 (0 MB)
[14:04:19][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
```

データが送信されていないと思うとこれは正しい挙動であるように思える。

Slicer

```
[14:04:17][STATE] READY ---> RUNNING
[14:04:17][INFO] daq::service::Plugin state : RUNNING
[14:04:17][DEBUG] daq::service::Plugin run number (from redis) = 200
[14:04:17][DEBUG] daq::service::MetricsPlugin state change: RUNNING
[14:04:17][INFO] fair::mq::Device running...
[14:04:18][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
```

```
[14:04:18][INFO] in[0]: in: 0 (0 MB) out: 0 (0 MB)
[14:04:18][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
[14:04:18][INFO] out[0]: in: 0 (0 MB) out: 0 (0 MB)
[14:04:19][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
[14:04:19][INFO] in[0]: in: 0 (0 MB) out: 0 (0 MB)
[14:04:19][DEBUG] daq::service::MetricsPlugin SendSocketMetrics done
[14:04:19][INFO] out[0]: in: 0 (0 MB) out: 0 (0 MB)
```

redisの中のkeyを全てflushしたところ正しく起動した。
セカンドフィルター内のデバック文を確認。

クラスターのデバック文も確認完了
TimeFrameメタは必ず後段に送信する必要がある。

- ~~FilterTimeFrameSliceByTOT.cxx~~
- ~~FilterTimeFrameSliceByTOT.h~~

- ~~FilterTimeFrameSliceByMultiplicity.cxx~~
- ~~FilterTimeFrameSliceByMultiplicity.h~~

- ~~FilterTimeFrame1SliceByTOF.cxx~~
- ~~FilterTimeFrameSliceByTOF.h~~

一旦分割した各種フィルターを作成した。
あとはトポロジーで繋ぎかえて測定できるか試してみる必要がある。

2024/7/15

0: TO DO

- トポロジー書き換えのためのドキュメント整備
- 各種フィルターをトポロジーで繋ぎ変える
- オプション
 - トラッキングフィルターの作成

一旦自分のリプレイヤーのディレクトリの方の整備は整った。

今度はaino-1上のnestdaq-user-implの方で整備を進める。

2024/7/17

0: TO DO

E585のデータ(RUN484)を解析し、オンラインフィルターにかけるゲートを決める。

現状では、6Li のゲートは次の通り

- 各面で3ヒット以上 (VDCのワイヤークラスターの話)
- Pla の1枚目で止まる。

チェックする項目は次の通り

- ヒットパターン分布
- TOT分布 (TOT - ΔE の対応表)

まずTOTでラフに6Liゲートを決めてその後ヒットパターンをヒストグラムに表示。

6Liが確かに3ヒット以上になっていることを確認する。

```
tree->Draw("vdc_gr_u2@.GetEntriesFast()", "vdc_gr_x1@.GetEntriesFast()>0 &&  
vdc_gr_u1@.GetEntriesFast()>0 && vdc_gr_x2@.GetEntriesFast()>0 &&  
vdc_gr_u2@.GetEntriesFast()>0 && pla_gr_att_1l@.GetEntriesFast()>0 &&  
pla_gr_att_1r@.GetEntriesFast()>0 && Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size() > 35 &&  
Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size() > 35")
```

これを6Liのゲートと考える。

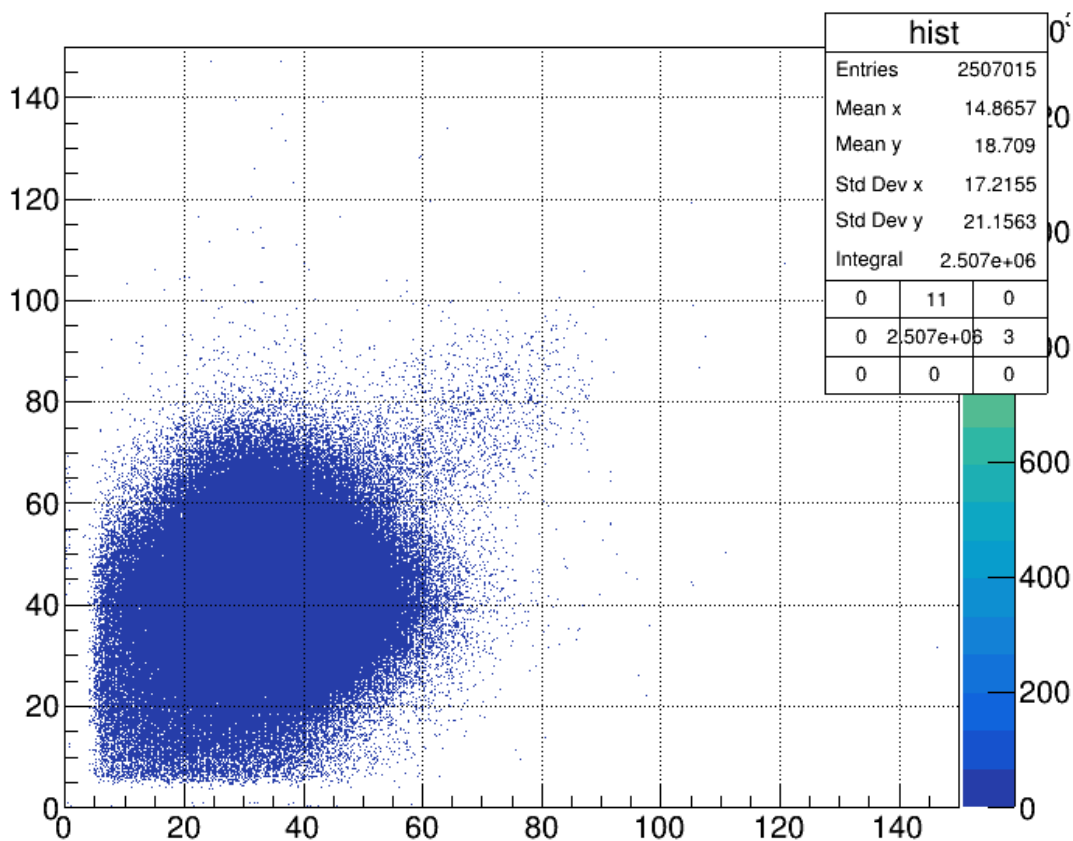
```
vdc_gr_x1@.GetEntriesFast()>0 && vdc_gr_u1@.GetEntriesFast()>0 &&  
vdc_gr_x2@.GetEntriesFast()>0 && vdc_gr_u2@.GetEntriesFast()>0 &&  
pla_gr_att_1l@.GetEntriesFast()>0 && pla_gr_att_1r@.GetEntriesFast()>0
```

 は、アクシデンタルを
取り除くためのゲート

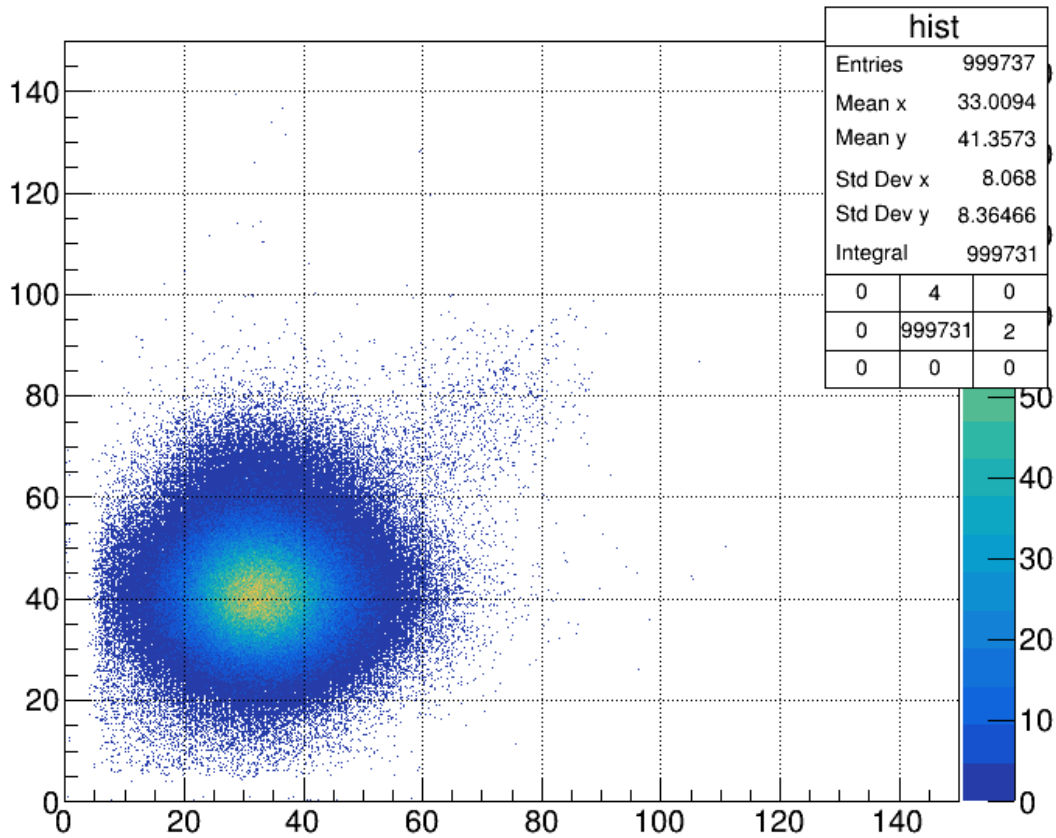
chkstrone_nishioka.yaml run0484 (2507015 evts recorded)

Wed Jul 17 14:57:29 2024(1721195849)

Sum\$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size():Sum\$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size()

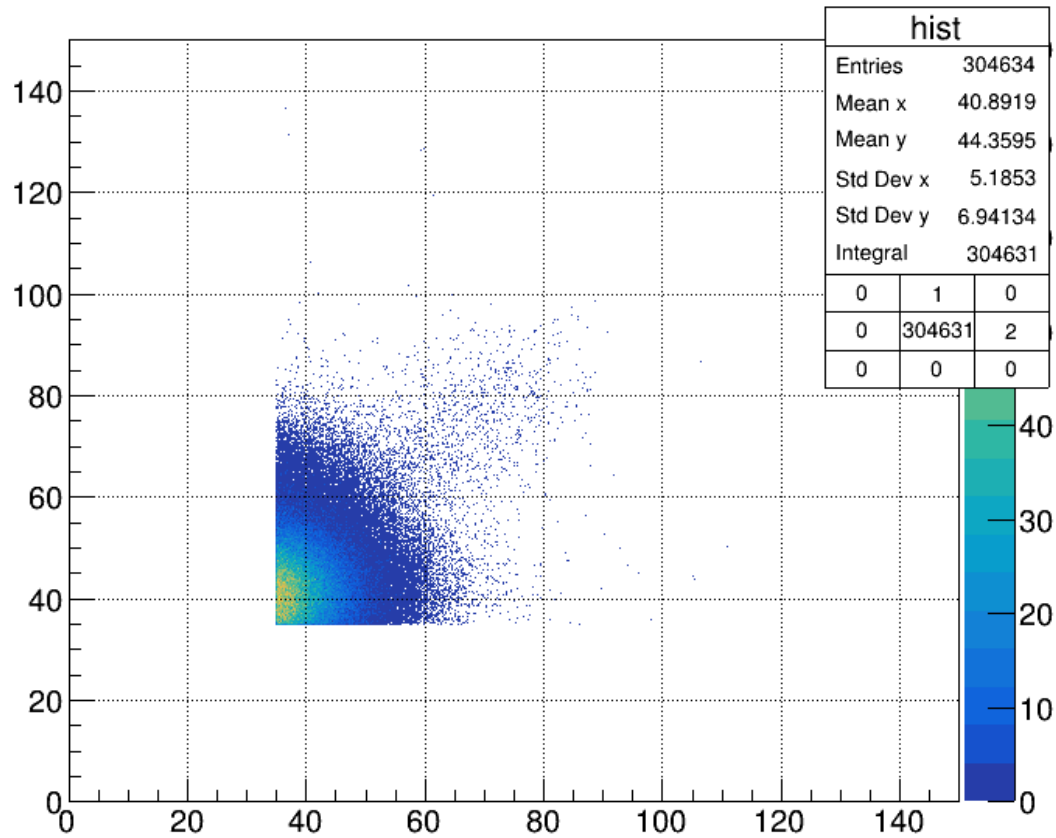


アクシデンタルを取り除くと次のヒストグラムになる。VDCとPLa1面のエントリ>0の時の条件になる。



TOTカットをかけた分布図

```
tree->Draw("Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size():Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size()>>hist(1000.,0., 150.,1000.,0.,150.)","vdc_gr_x1@.GetEntriesFast()>0 &&
vdc_gr_u1@.GetEntriesFast()>0 && vdc_gr_x2@.GetEntriesFast()>0 &&
vdc_gr_u2@.GetEntriesFast()>0 && pla_gr_att_1l@.GetEntriesFast()>0 &&
pla_gr_att_1r@.GetEntriesFast()>0 && Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size() > 35 &&
Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size() > 35","colz")
```

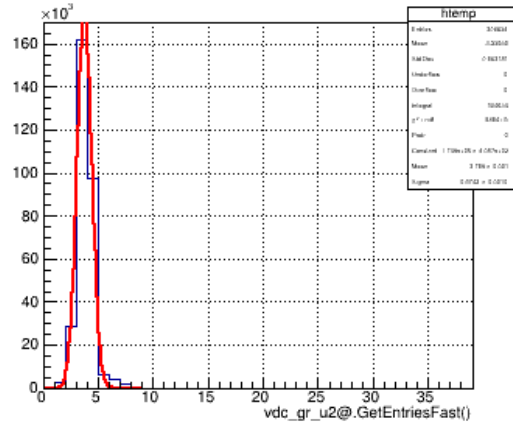
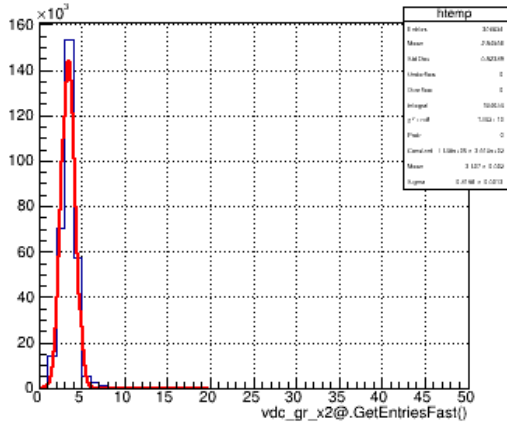
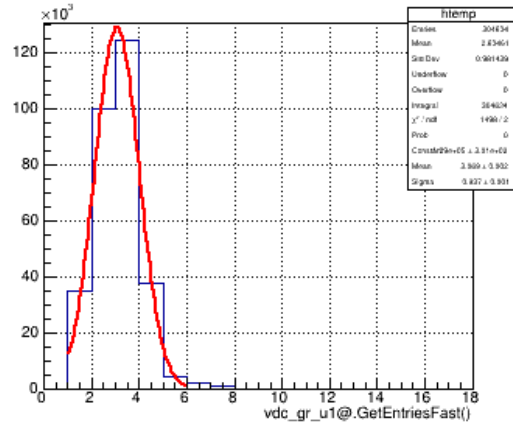
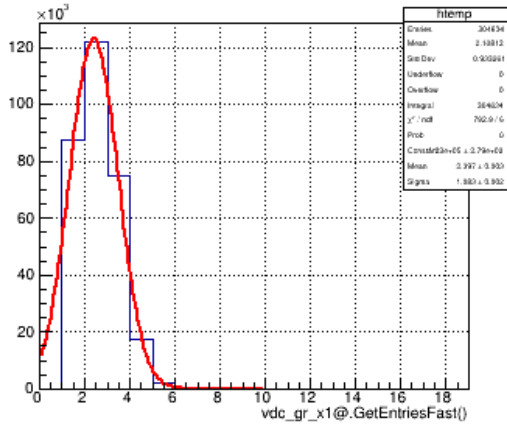


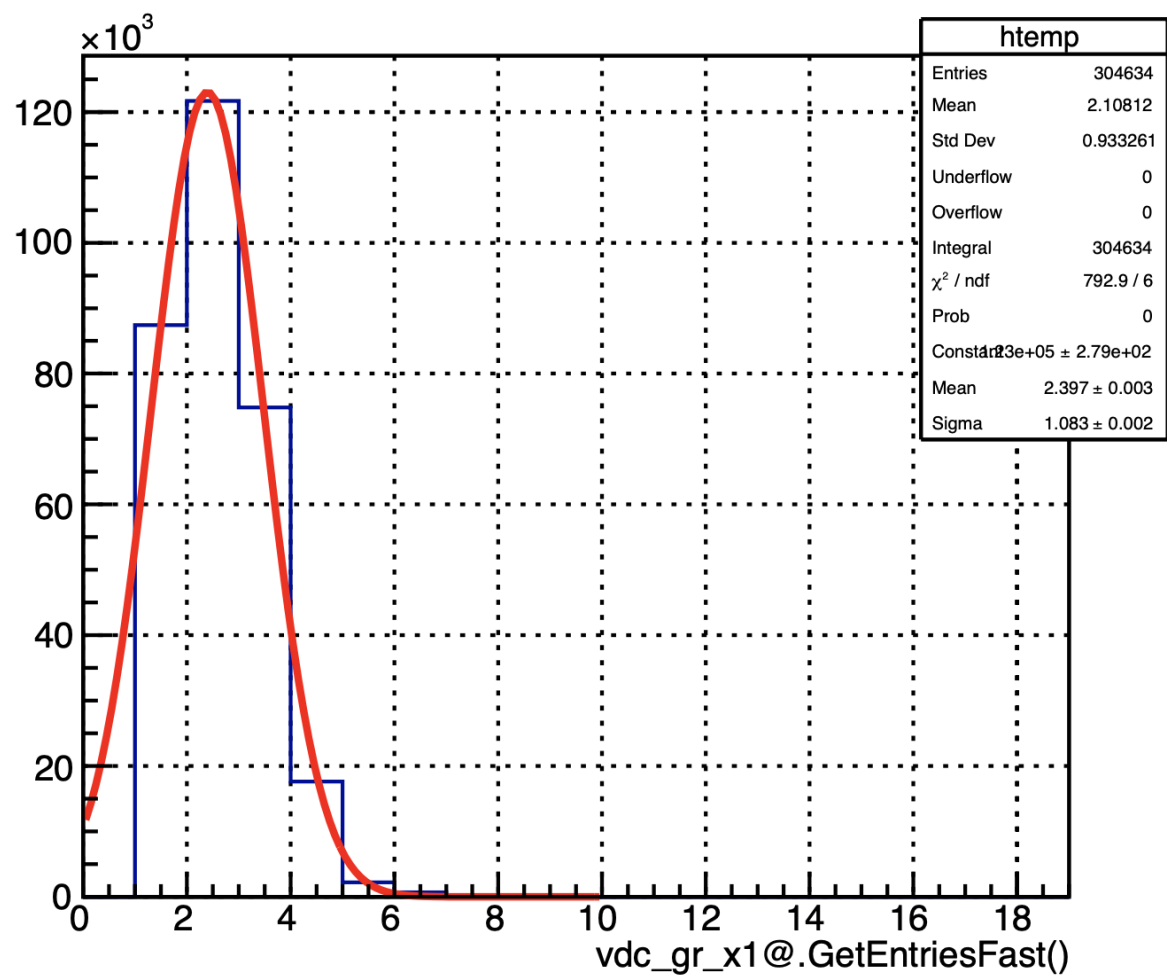
この時のVDCのヒット分布は次の通りになる。

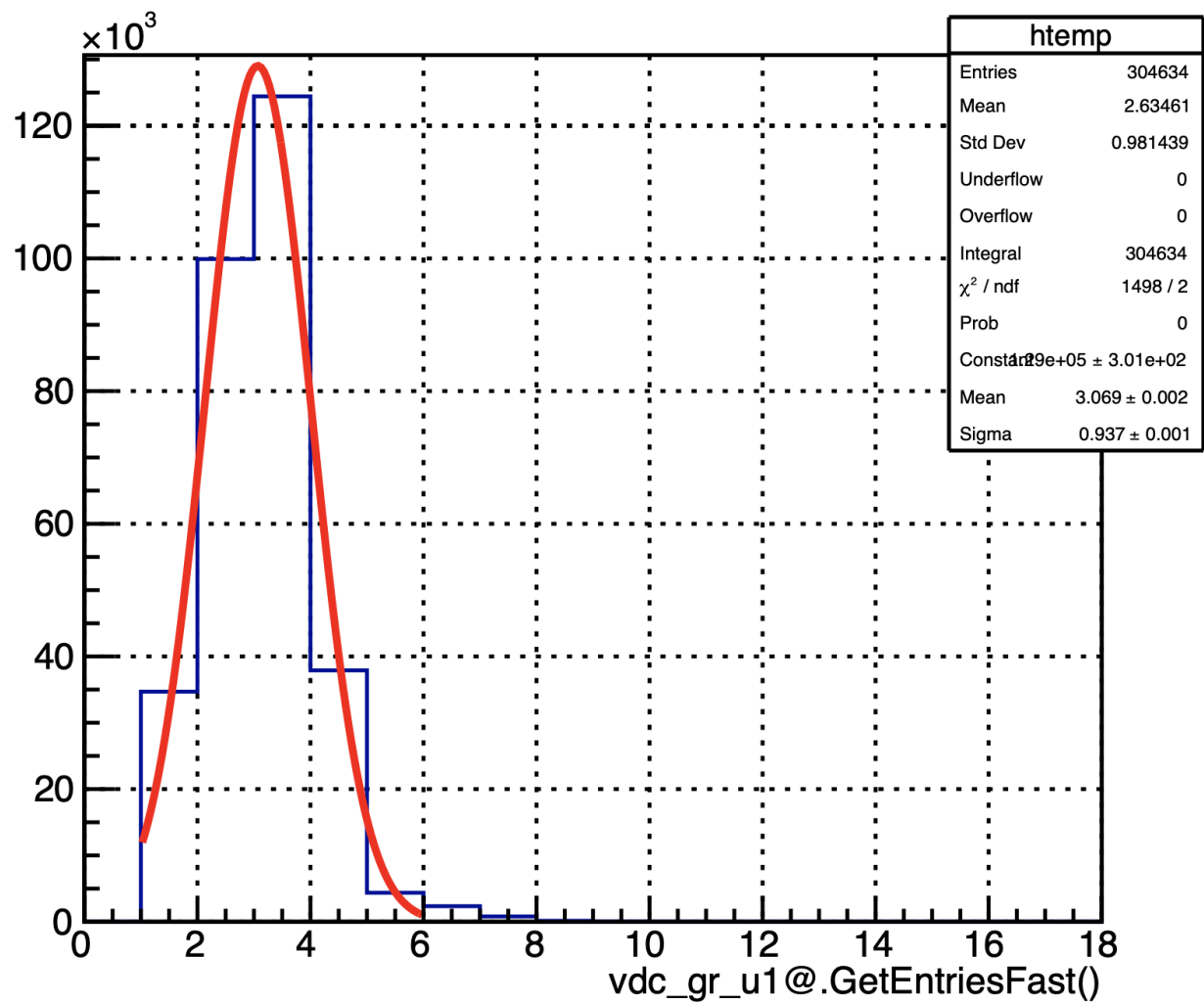
```
"vdc_gr_x1@.GetEntriesFast(>0 && vdc_gr_u1@.GetEntriesFast(>0 &&
vdc_gr_x2@.GetEntriesFast(>0 && vdc_gr_u2@.GetEntriesFast(>0 &&
pla_gr_att_1l@.GetEntriesFast(>0 && pla_gr_att_1r@.GetEntriesFast(>0 &&
Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size() > 35 &&
Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size() > 35"
```

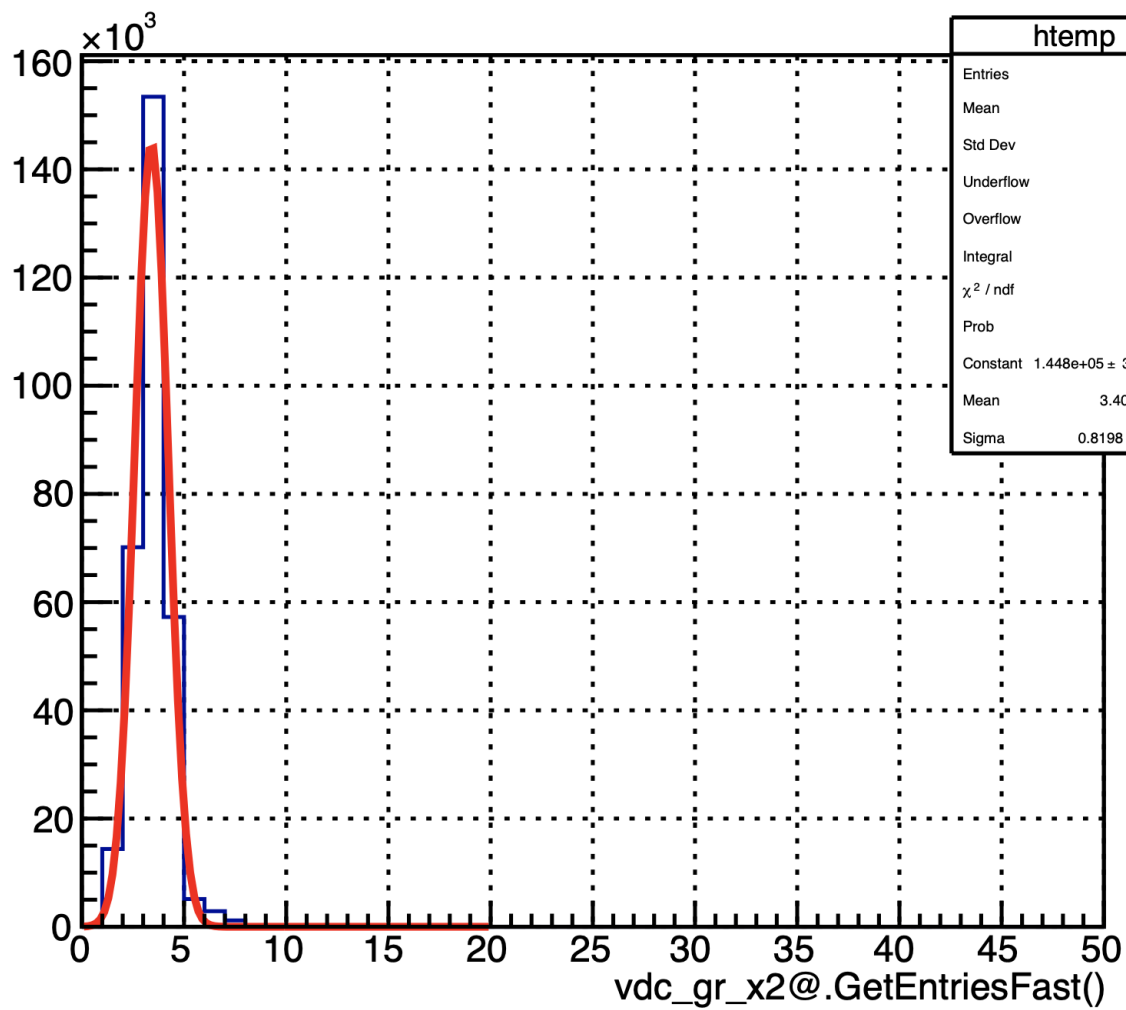
chkstrone_nishioka.yaml run0484 (2507015 evts recorded)

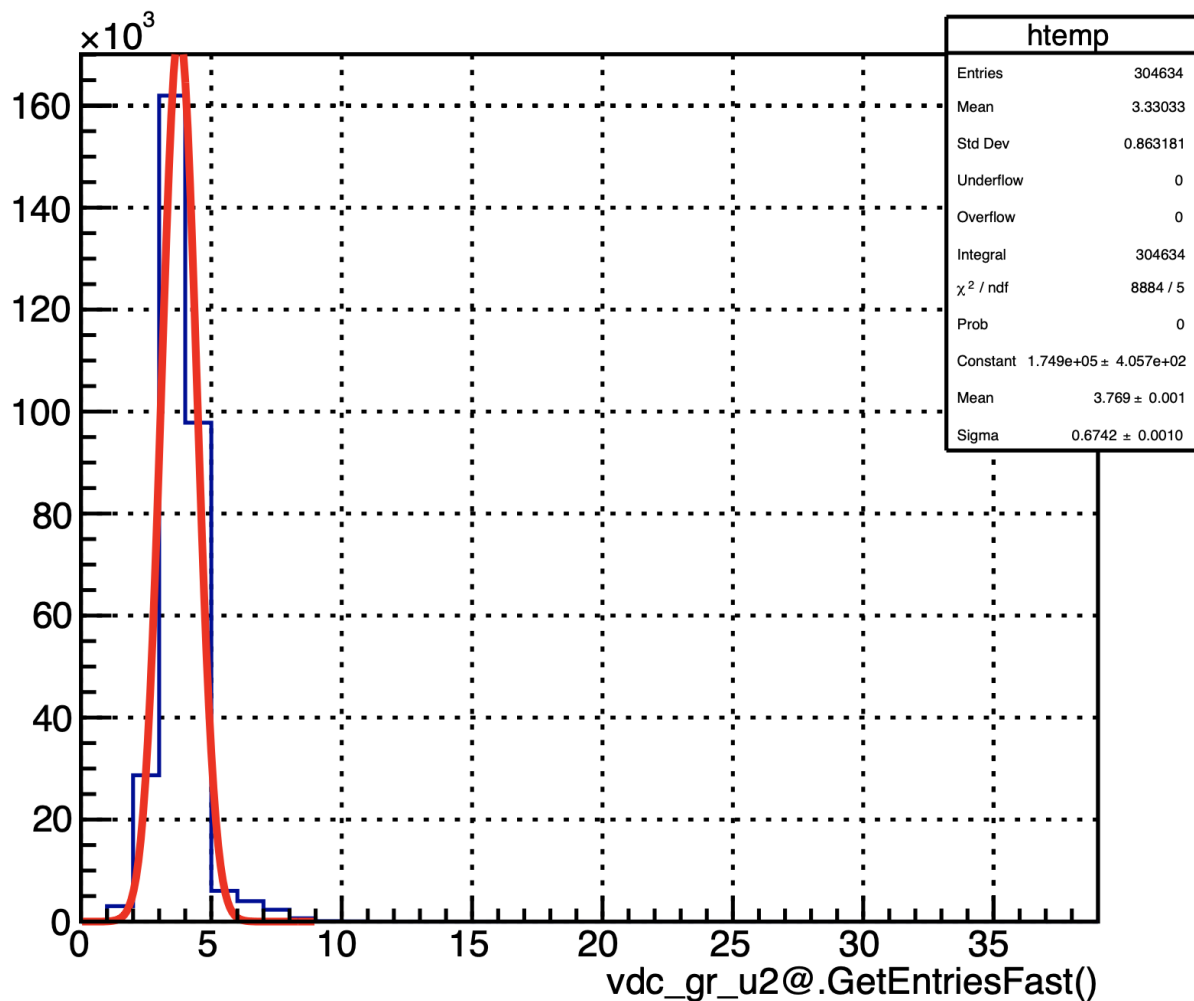
Wed Jul 17 15:34:25 2024(1721198065)











x1以外全て3にピークを持っていた。

x1: 2.79

u1: 3.069

x2: 3.047

u2: 3.769

クラスターの定義: 各面で2ヒット以上ある時

2024/7/19 9:27

FilterTimeFrameByMultiplicityに不具合を見つけた。

FilterTimeFrameABC.icxx の Initialize でvdcのチャンネルマップをロードしている。

#CATID	ID	sh	fcp	det	GEO	CH
51	0	0	0	0	0xc0a802a1	0

変数が7つ、これがチャンネル数だけ存在するので、 $7 * 800 = 5600$ 個
1変数 4bytと思うと、 $4 * 5600 = 22400$ byt

チャンネル数増加に伴いメモリを食うのでLoadingwiremapの部分だけ読み込む部分を変えたのを忘れていた。

Wire_map details - catid: -1, id: -1, geo: 18446744073709551615
Wire_map catid is -1 for geoIndex: 6, channel: 1

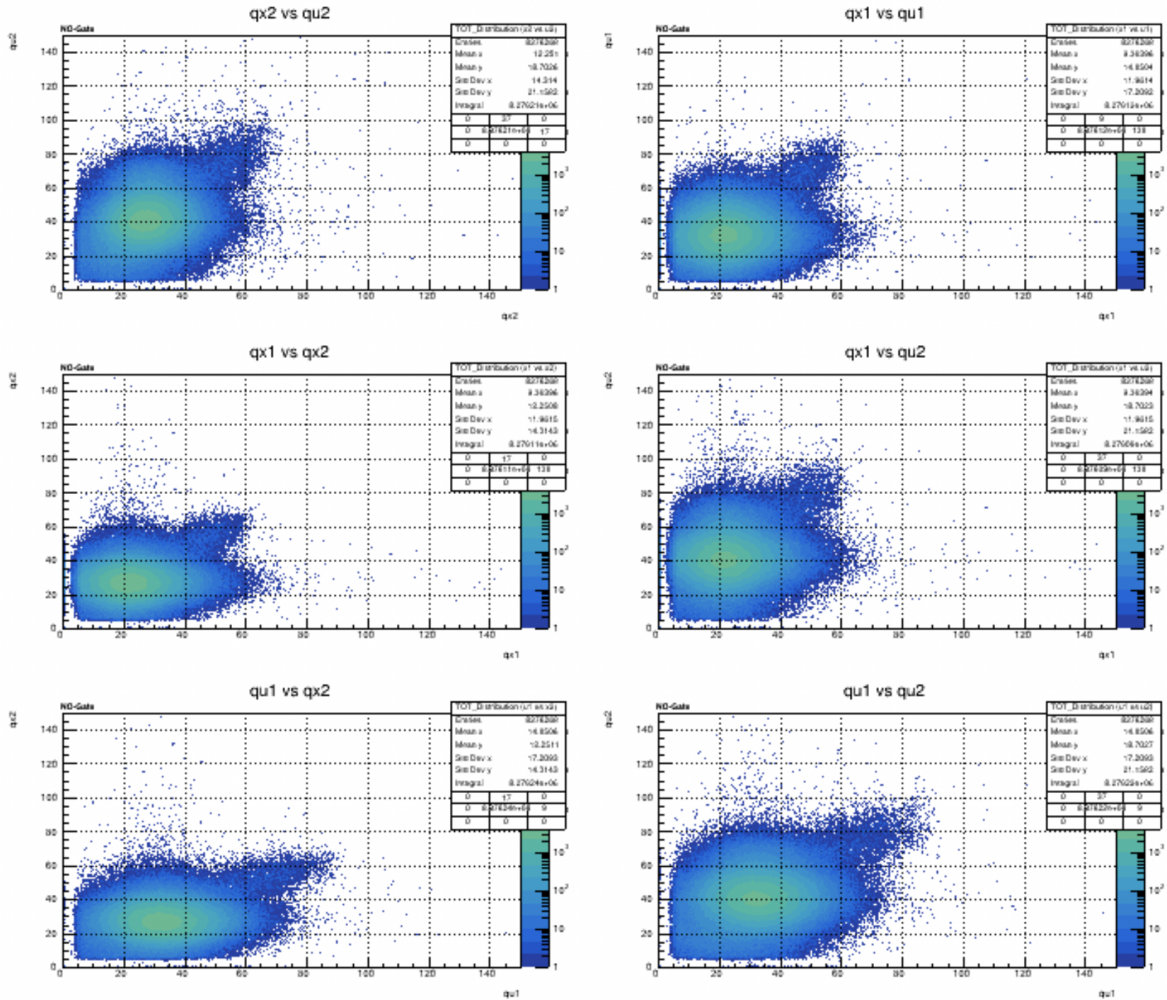
→前回の全て読み込む方式に変更

Loadingwiremapにデバック文を追加。
→規定クラスにGeoToIndexを追加し直して修正完了。

デバック文を見る限りでは正しく動いていそう。

2024/7/21

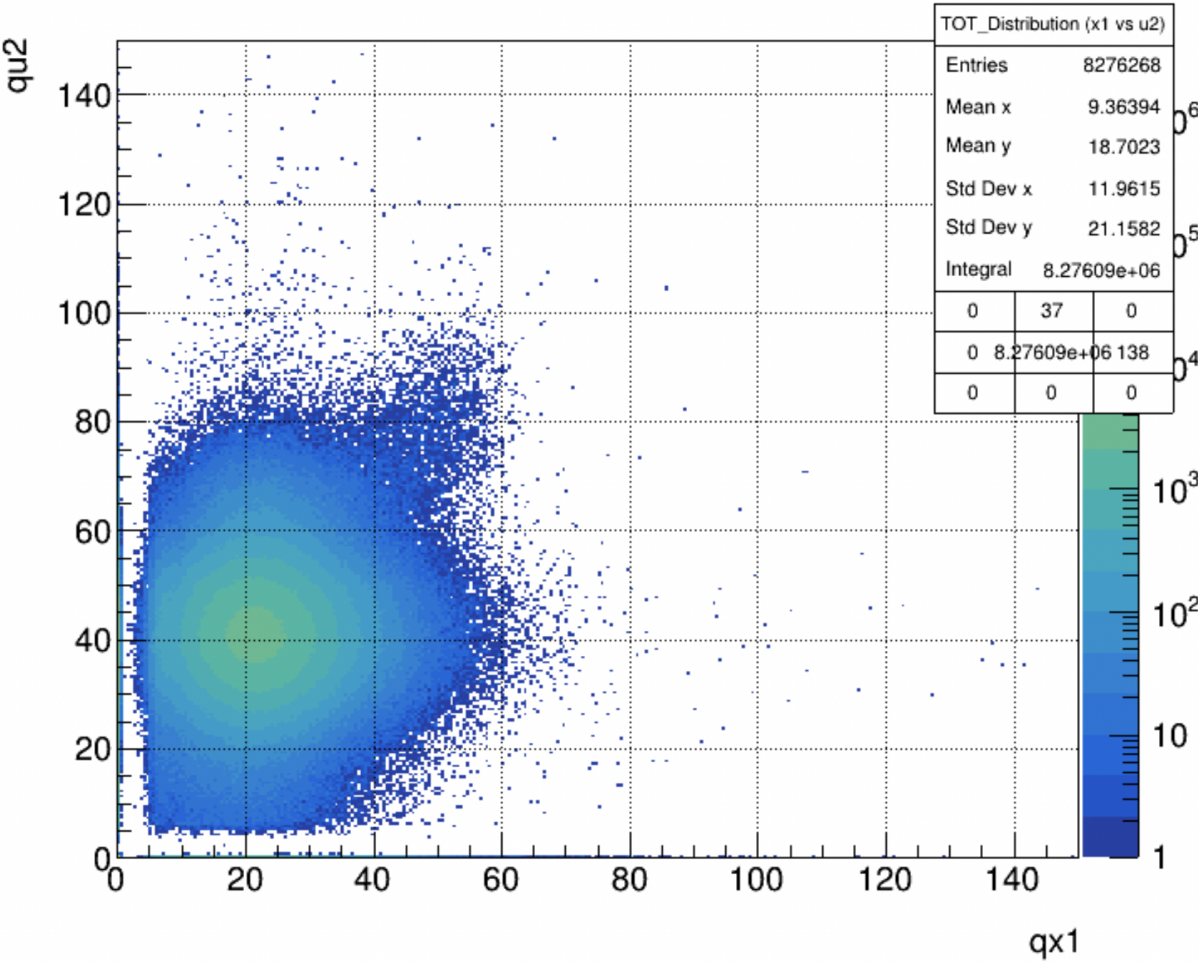
TOTフィルターの閾値決め



それぞれ1つ1つ見ることにする。

- qx2 vs qu2
- qx1 vs qx2
- qu1 vs qx2
- qx1 vs qu1
- qx1 vs qu2
- qu1 vs qu2
- qx2 vs qu2

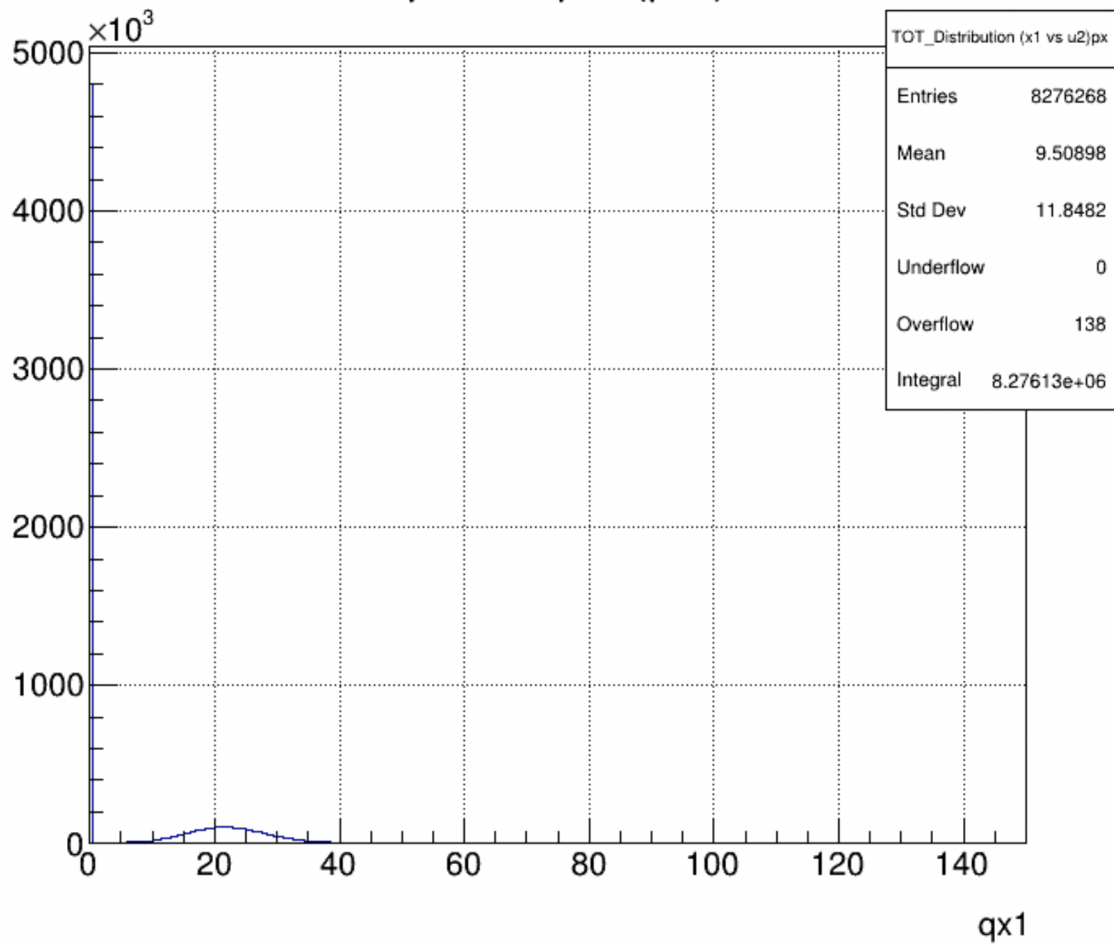
qx1 vs qu2



chkstrone.yaml run0484 (8276268 evts recorded)

Sun Jul 21 17:41:31 2024(1721551291)

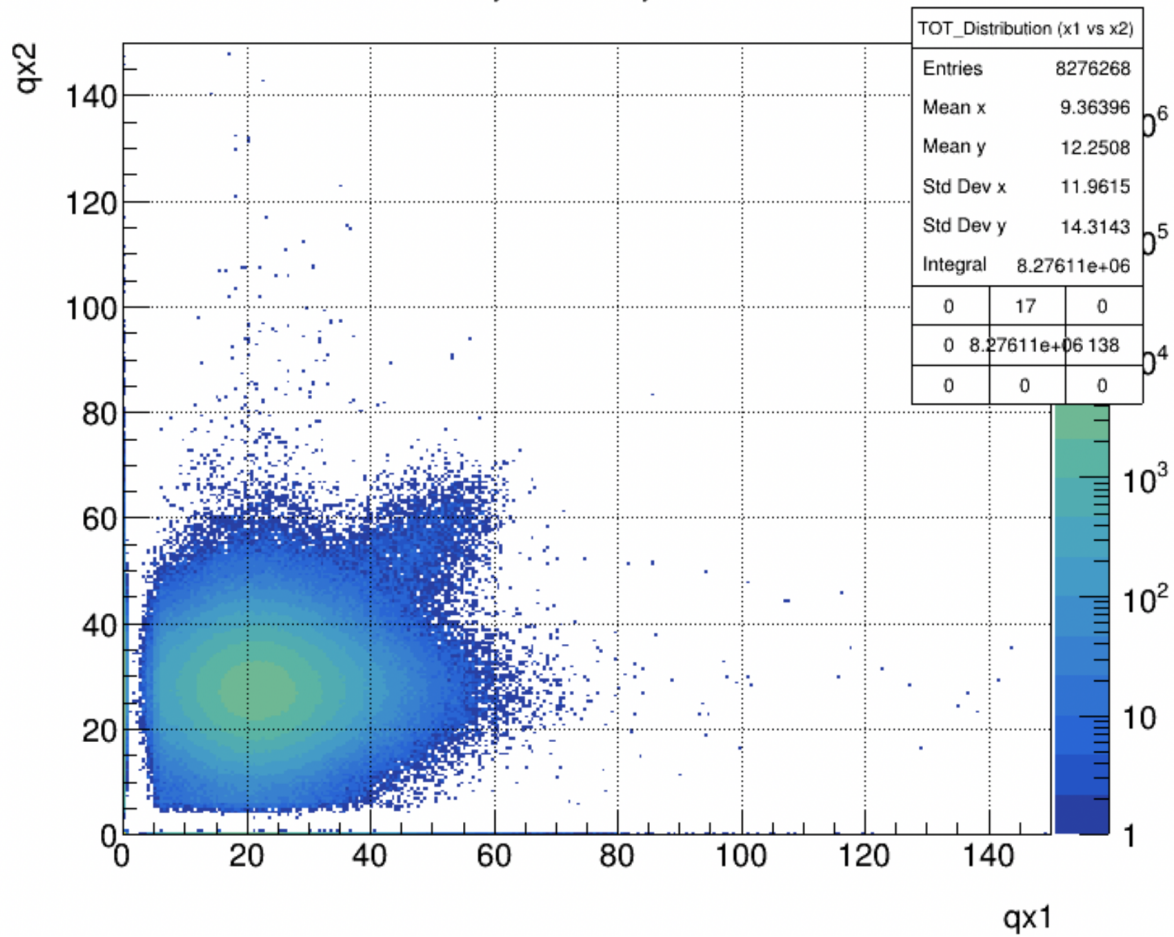
qx1 vs qu2 (prx)



ここからラフにゲートは
 $pu2 > 50 \ \&\& \ qx1 > 30$ と決める。

- qx1 vs qx2

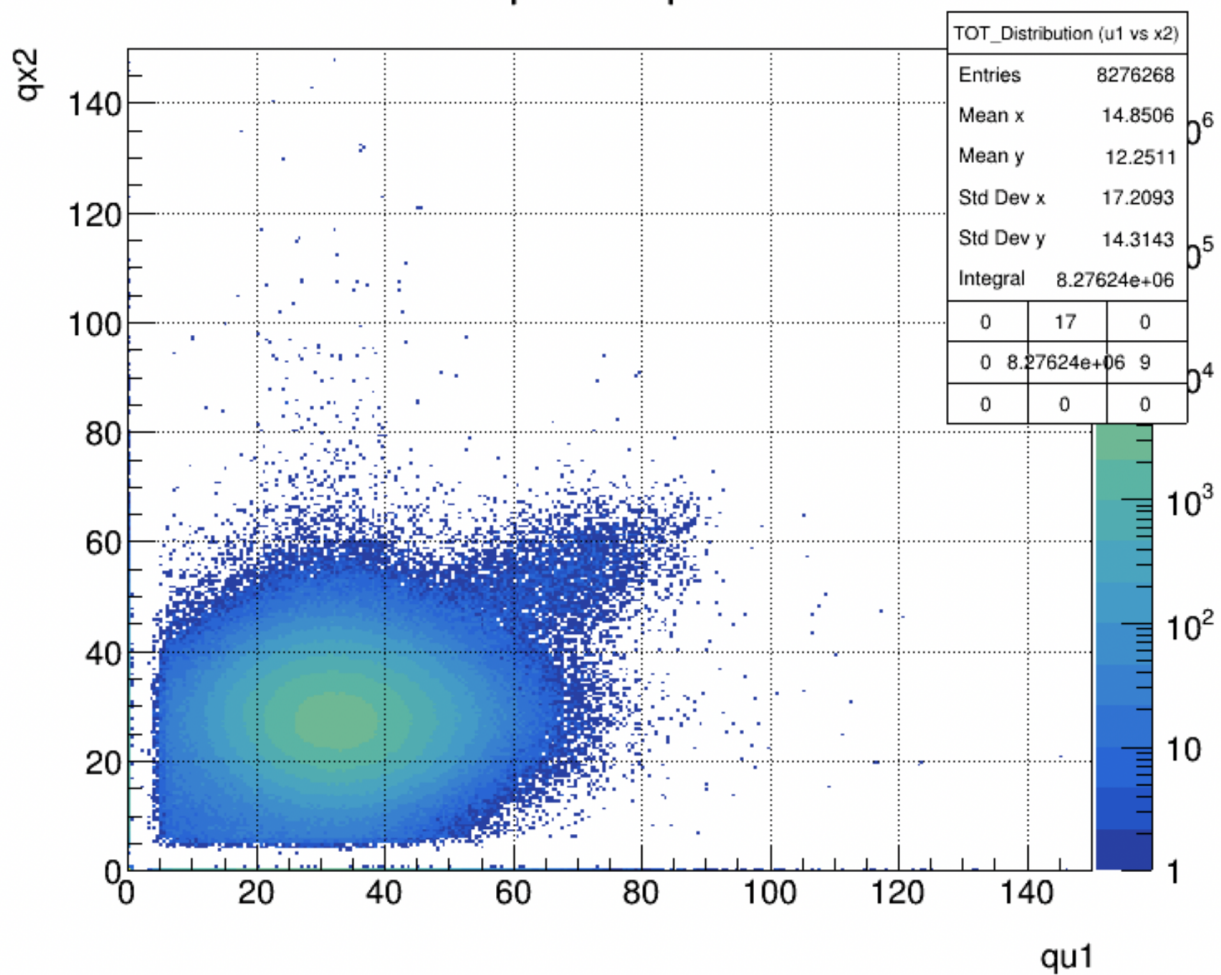
qx1 vs qx2



qx1 > 30 && qx2 > 30

- qu1 vs qx2

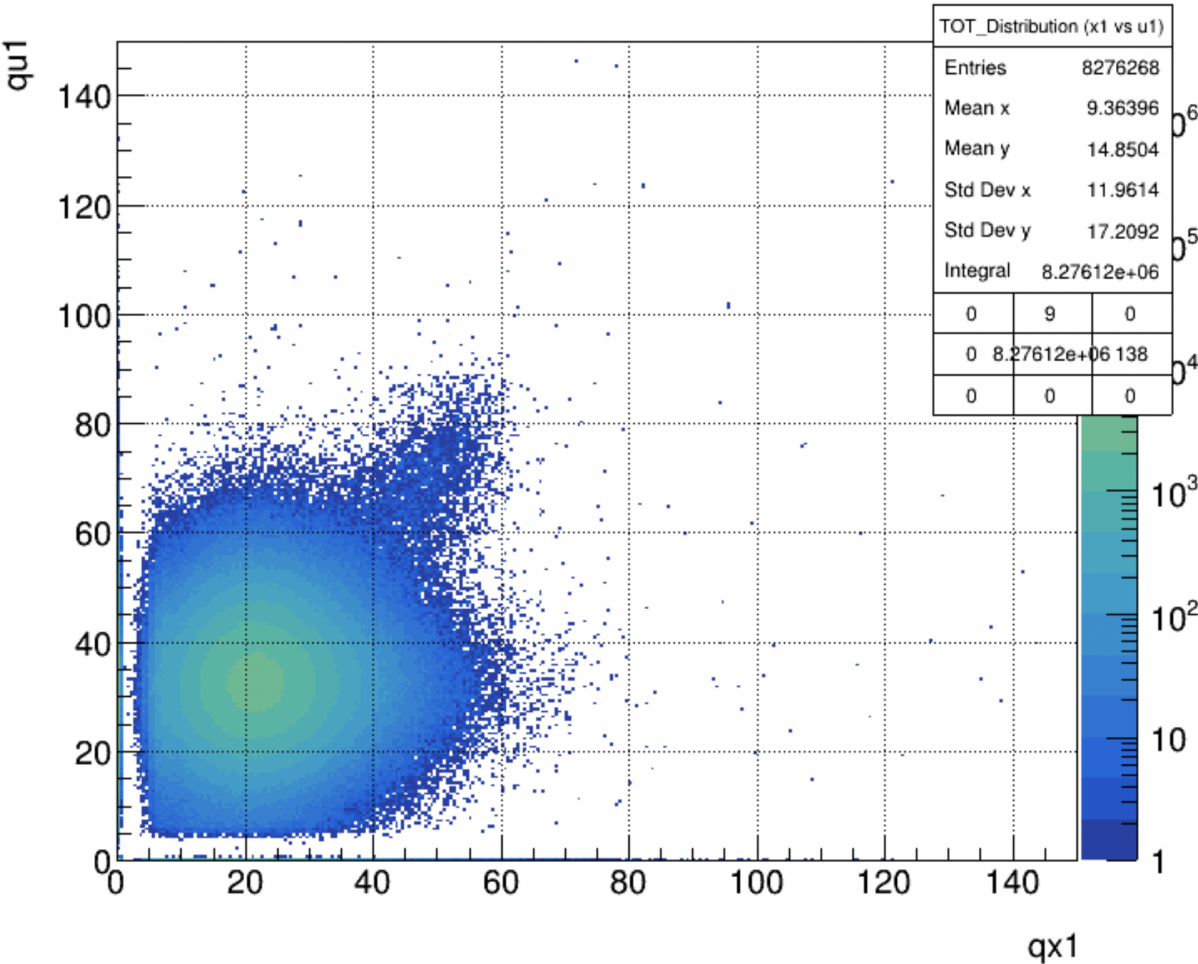
qu1 vs qx2



qu1 > 40 && qx2 > 30

- qx1 vs qu1

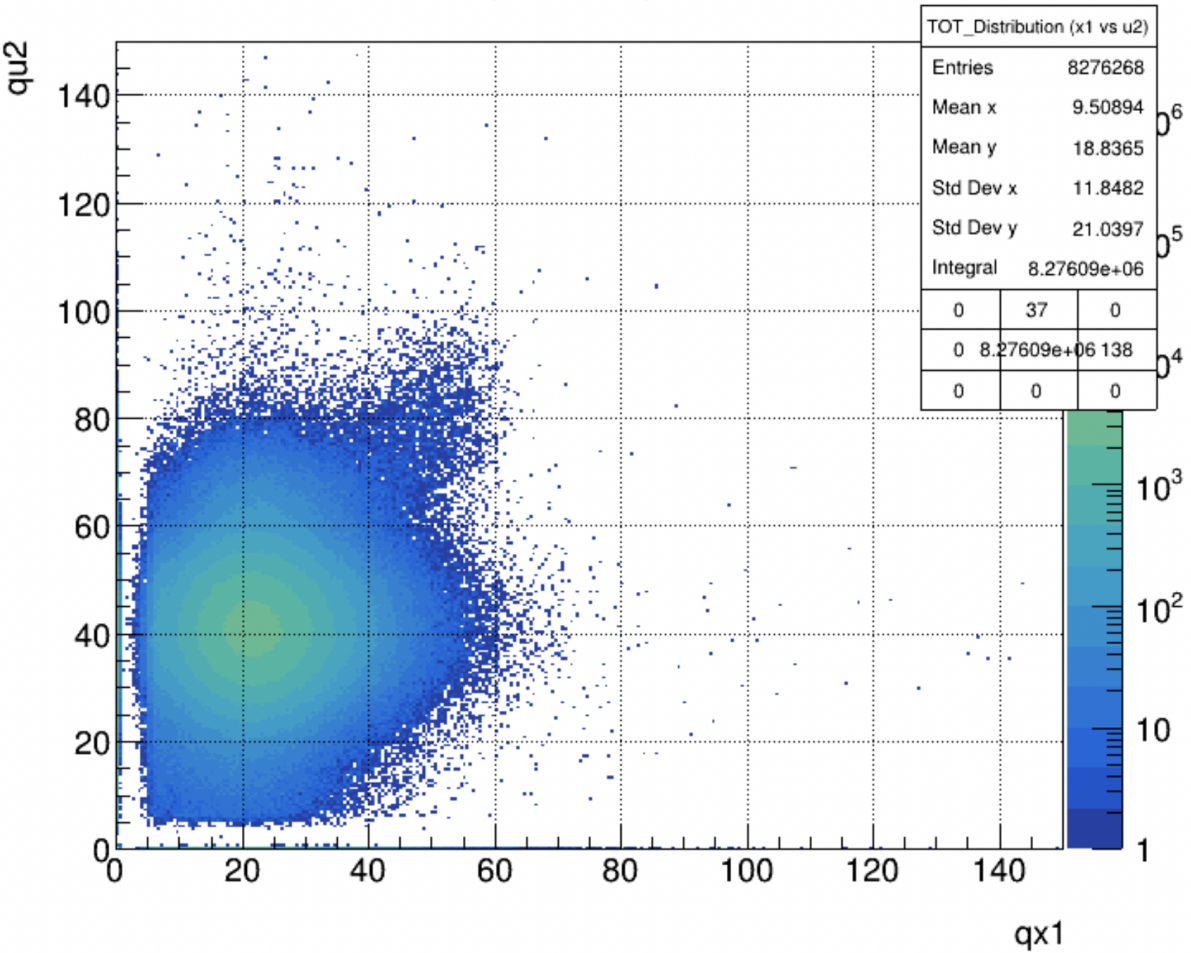
qx1 vs qu1



qu1 > 40 && qx1 > 20

- qx1 vs qu2

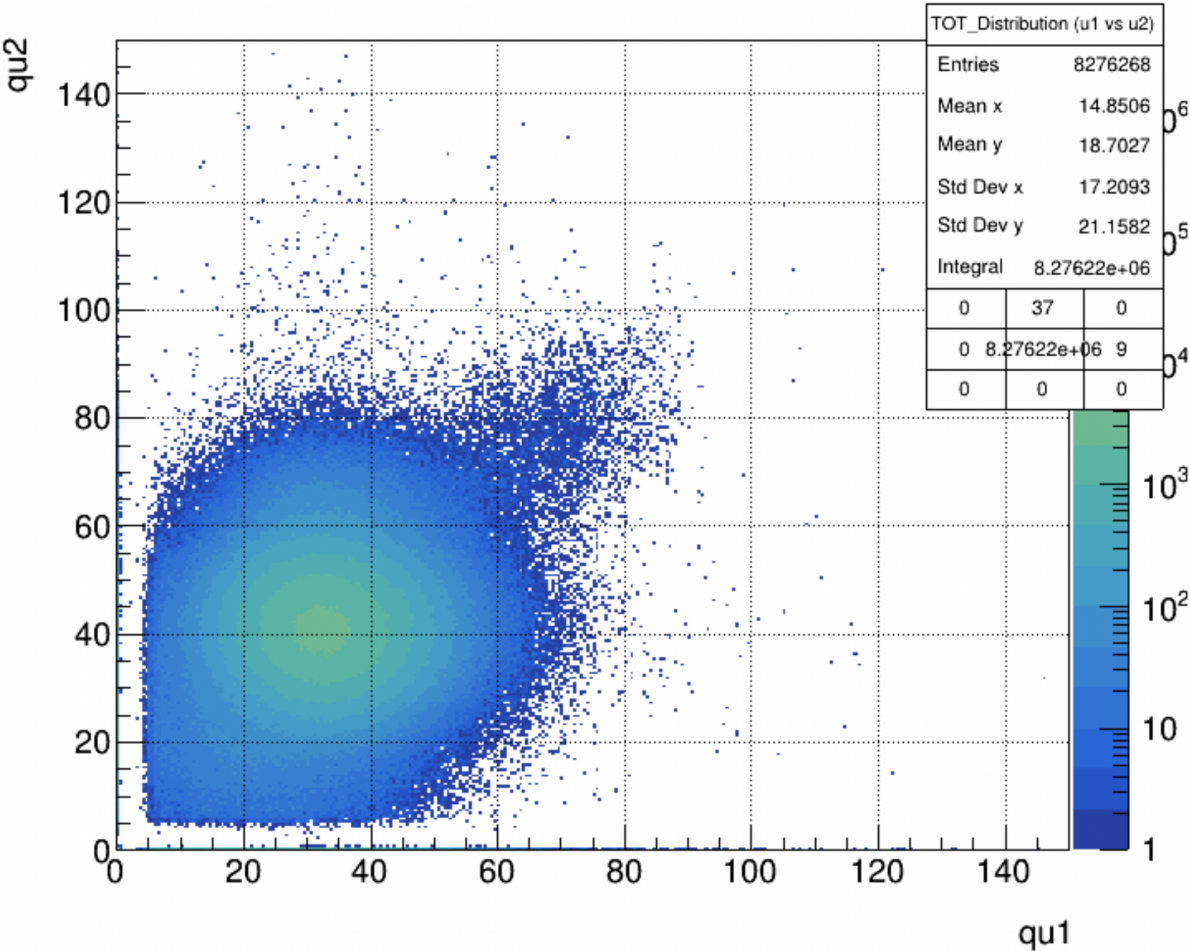
qx1 vs qu2



qu2 > 50 && qx1 > 25

- qu1 vs qu2

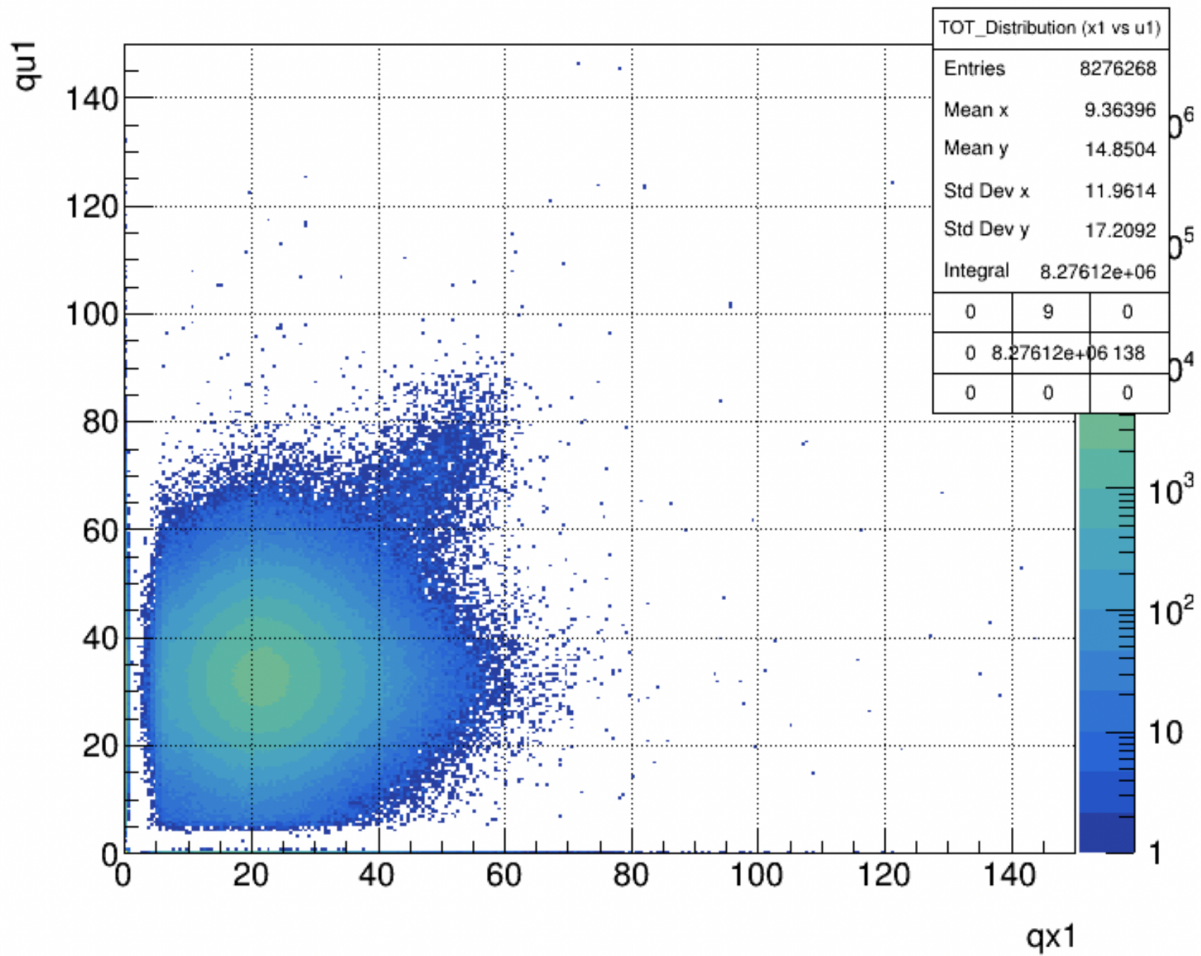
qu1 vs qu2



qu1 > 40 && qu2 > 40

- qx2 vs qu2

qx1 vs qu1



$qx1 > 30 \ \&\& \ qu1 > 30$

ゲートまとめ

$qu2 > 50 \ \&\& \ qx1 > 30$

$qx1 > 30 \ \&\& \ qx2 > 30$

$qu1 > 40 \ \&\& \ qx2 > 30$

$qu1 > 40 \ \&\& \ qx1 > 20$

$qu2 > 50 \ \&\& \ qx1 > 25$

$qu1 > 40 \ \&\& \ qu2 > 40$

つまり

```
qx1 > 20  
qx2 > 30  
qu1 > 40  
qu2 > 40
```

この条件でゲートをかける。

```
tree->Draw("Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size():S  
um$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size()>>hist(1000.,0.  
, 150.,1000.,0.,150.)", "vdc_gr_x1@.GetEntriesFast()>0  
&& vdc_gr_u1@.GetEntriesFast()>0 &&  
vdc_gr_x2@.GetEntriesFast()>0 &&  
vdc_gr_u2@.GetEntriesFast()>0 &&  
pla_gr_att_1l@.GetEntriesFast()>0 &&  
pla_gr_att_1r@.GetEntriesFast()>0 &&  
Sum$(vdc_gr_x1.fCharge)/vdc_gr_x1@.size() > 20  
&&Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size() > 40  
&&Sum$(vdc_gr_x2.fCharge)/vdc_gr_x2@.size() > 30 &&  
Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size() > 40"  
, "colz")
```

→ このゲートを使って、ヒット分布を見してみる。

オフライン解析のTOTと実際のTOT分布を比較して、値を対応づける必要がある。→ 前回のnestdaq-user-implを利用して、TOT分布を出力してみる。

RUN484 dcmdataのリプレイをする必要がある。

→ topo_player.sh

→ run.sh

2023/7版に変更

TFBFilePlayer -> EventBuilder -> fltcoin -> Sink

fltcoin で直接TOT値をダンプしてみる。
ファイル名「charge_ratio.txt (RUN484) 」

オンラインの性能評価

run000350.dat を利用して、オンラインノード数の評価を行う。

- フィルターの性能評価
 - TOT フィルター
 - Multiplicity フィルター
 - (Tracking フィルター)
 - (TOFフィルタ)

絶対に評価する必要のあるフィルターは2つ。
TOTフィルターとMultiplicityフィルター、である。
評価項目は次の通り。

- 単位時間あたりにどれくらいのデータを処理できたか？(μsあたりの話)
- リダクションレート→ これはゲート依存、真に評価すべきは本物を落としていないかどうかの確認。
 - フィルターの前後で本物らしきピークを削っていないか確認。

最小要件 (6Liゲート)

各VDC面のエントリ>0 && Pla1枚目のエントリー

斜めにプロジェクションできないかどうか？

→ヒストグラムに詰めた後に傾けるとビン幅が歪むので、フィルする前に傾ける。

本物を落としたくないと思うと信頼区間99%(5σ) でゲートをかけたいと思う。
ミニマムバイアス(各面のエントリーが0より大きいとき)

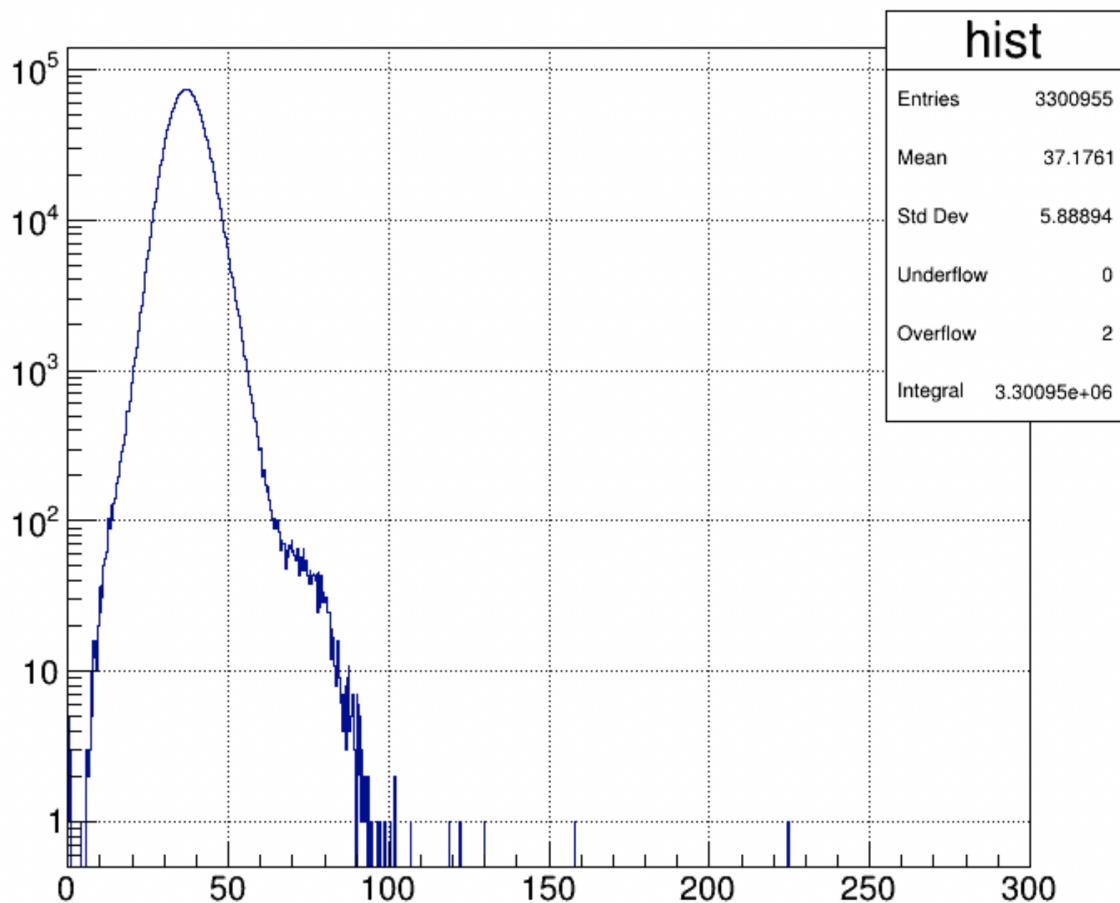
```
tree->Draw("((Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size())
```

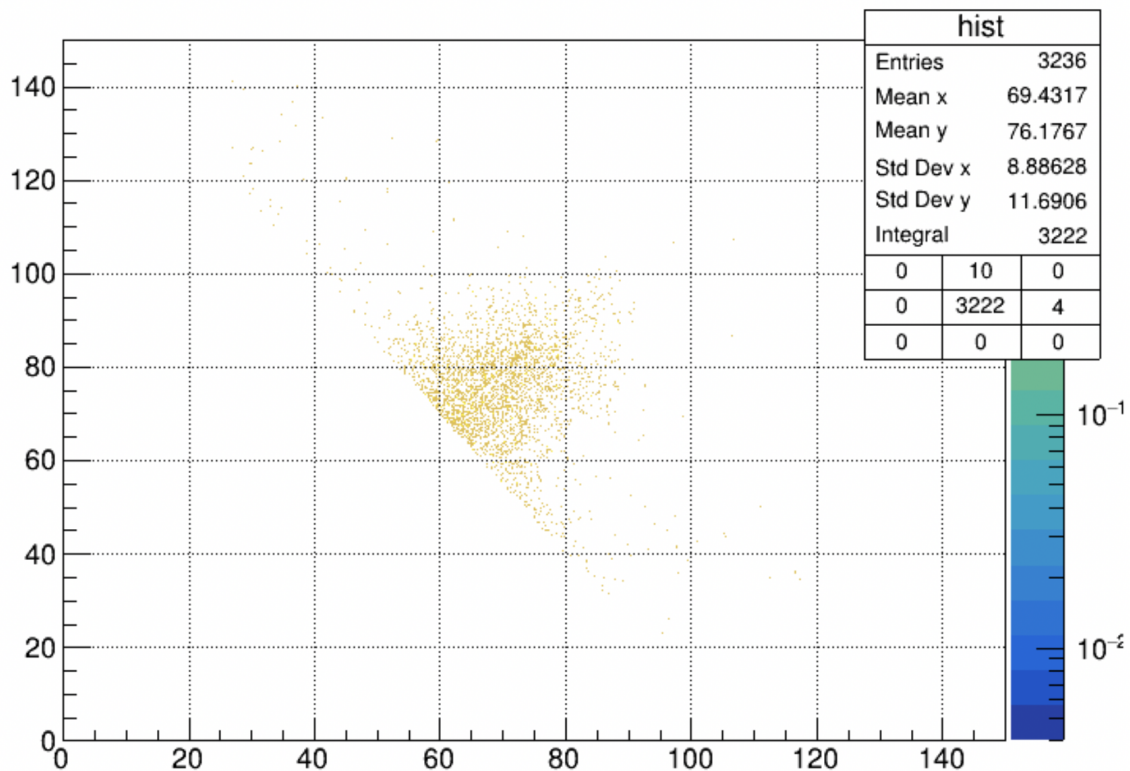
```
)+(Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size()))/2.>>hist
(1000.,0., 300.)", "vdc_gr_x1@.GetEntriesFast()>0 &&
vdc_gr_u1@.GetEntriesFast()>0 &&
vdc_gr_x2@.GetEntriesFast()>0 &&
vdc_gr_u2@.GetEntriesFast()>0 &&
pla_gr_att_1l@.GetEntriesFast()>0 &&
pla_gr_att_1r@.GetEntriesFast()>0", "colz")
```

45度傾けたいと思うと、ヒストグラムにフィルする前にブランチを45度傾ける必要がある。

chkstrone.yaml run0484 (8276268 evts recorded)

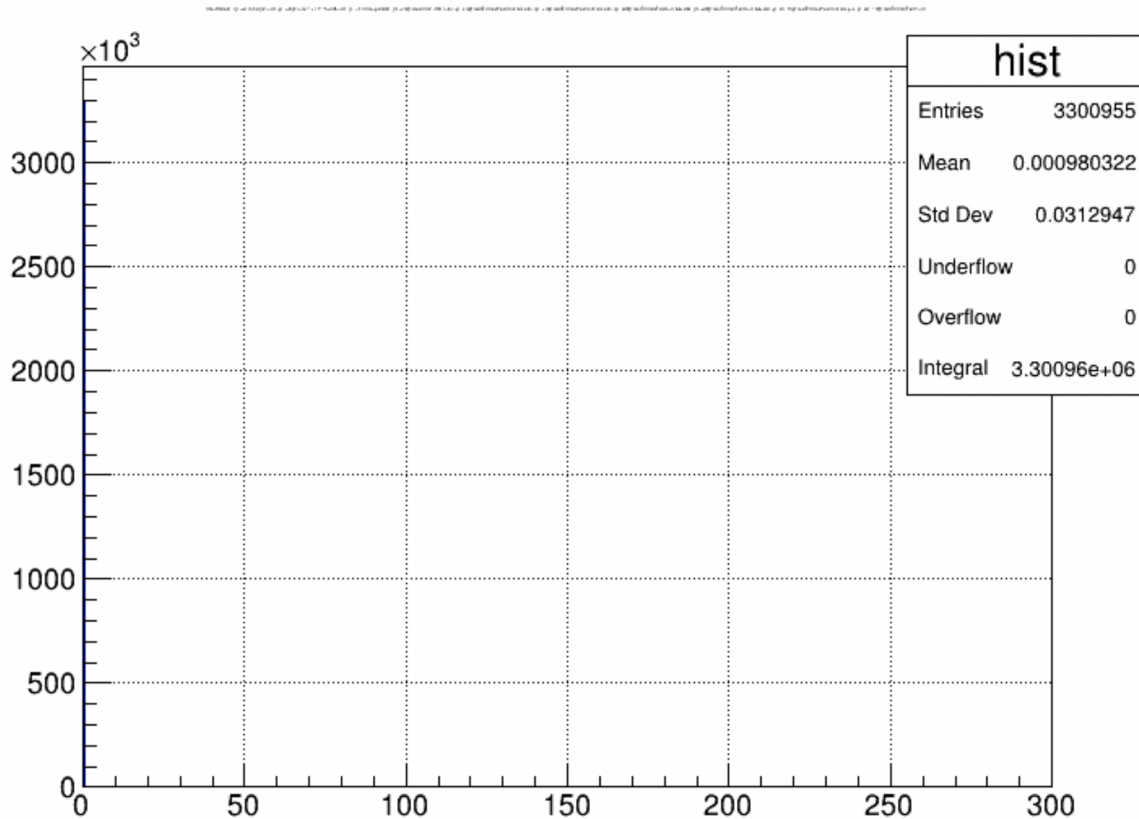
Sun Jul 21 21:30:01 2024(1721565001)





↑ カットの位置はただしそう。

```
artemis [23]
tree->Draw("Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size():S
um$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size()>>hist(1000.,0.
, 150.,1000.,0.,150.)","vdc_gr_x1@.GetEntriesFast()>0
&& vdc_gr_u1@.GetEntriesFast()>0 &&
vdc_gr_x2@.GetEntriesFast()>0 &&
vdc_gr_u2@.GetEntriesFast()>0 &&
pla_gr_att_1l@.GetEntriesFast()>0 &&
pla_gr_att_1r@.GetEntriesFast()>0 &&
((Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size())+1.5*(Sum$(
vdc_gr_u1.fCharge)/vdc_gr_u1@.size()))/2.>80.", "colz")
```



↑

```

artemis [23]
tree->Draw("((Sum$(vdc_gr_u2.fCharge)/vdc_gr_u2@.size())
)+(Sum$(vdc_gr_u1.fCharge)/vdc_gr_u1@.size()))/2.>>hist
(1000.,0., 300.)","vdc_gr_x1@.GetEntriesFast()>0 &&
vdc_gr_u1@.GetEntriesFast()>0 &&
vdc_gr_x2@.GetEntriesFast()>0 &&
vdc_gr_u2@.GetEntriesFast()>0 &&
pla_gr_att_1l@.GetEntriesFast()>0 &&
pla_gr_att_1r@.GetEntriesFast()>0 &&
","colz")

```

本物の割合 (違う手段で調べる必要がある。)

ゲート

- 本物を落としていないかどうか？
 - 本物を落とさずにどれだけゴミを落とせるか？
 - 違う手段で「本物がどれだけいるか」調べる必要がある。
 - ゲートはチェック
- 削減率
 - ゲート後、何%本物がいるか？
 - ガウシアンフィット→ガウシアンของフィッティングパラメータの誤差を評価する。
 - パラメータ空間での誤差評価

新しいバージョンのNestDAQに変更する。

オンライン実行の性能評価

Trackingフィルターのデバックを進める。