

EXPERT ROOT - Developing #210

DigiBuilder

07/11/2018 09:20 AM - Vitaliy Schetinin

Status:	Закрыта	Start date:	07/11/2018
Priority:	Низкий	Due date:	
Assignee:	Vitaliy Schetinin	% Done:	100%
Category:	beamtime	Estimated time:	0.00 hour
Target version:	v-1.0		

Description

DigiBuilder

DigiBuilder является интерфейсом для чтения файлов экспериментальных данных - результатов работы библиотеки https://github.com/evovch/ACCULINNA_go4_user_library.

А точнее файлов, появляющихся после этапа Calibration:

https://github.com/evovch/ACCULINNA_go4_user_library/blob/master/docu/DAQ_ER_diagram2.pdf.

На этом этапе данные представлены в виде root дерева, в ветках которого лежат объекты класса CalDetMessage .

В рамках этой задачи предлагается научить DigiBuilder работать с CalDetMessage только от телескопов и бимдет. Остальные детекторы будут добавлены в последующих задачах.

Надо обкатать схему.

DigiBuilder должен быть наследником FairSource и реализован по примеру

<https://github.com/ExpertRootGroup/er/blob/dev/beamtime/ERRootSource.h> .

Простое чтение событий из внешнего файла с помощью fTree->GetEntry();

Запуск происходит с помощью FairRunOnline.

В результате работы должен получаться root файл в ветках которого лежат диджи.

На данный момент для каждого детектора реализован свой диджи, более того, реализованы диджи для разных типов станций детектора (для телескопа отдельно для Si и отдельно для Csl):

<https://github.com/ExpertRootGroup/er/blob/dev/telescope/data/ERQTelescopeCslDigi.h>

<https://github.com/ExpertRootGroup/er/blob/dev/telescope/data/ERQTelescopeSiDigi.h>

<https://github.com/ExpertRootGroup/er/blob/dev/BeamDet/data/ERBeamDetMWPCDigi.h>

<https://github.com/ExpertRootGroup/er/blob/dev/BeamDet/data/ERBeamDetTOFDigi.h>

Такой подход живет с давних времени и вызван тем, что адресация digi реализована по разному для разных типов станций и хранится в самом диджи:

К примеру для ToF это fToFNb, а для MWPC это (fMWPCNb + fPlaneNb + fWireNb)

На данный момент для большинства детекторов мы от такого подхода уходим и вся адресация digi, кроме номера канала внутри станции хранится в названии ветки. см Рис.

В связи с этим будет введен один базовый класс диджи содержащий в себе время и амплитуду, и в редких случаях будут создаваться его наследники.

Структура классов еще не переделана.

ВНИМАНИЕ: Не надо ее переделывать в рамках этой задачи. Это будет сделано сразу во всем репозитории в рамках другой задачи и ветки.

В рамках данной задачи для каждого типа детектора и станции в нем в ветку надо складывать объект соответствующего класса. Тем более, что этот механизм нам пригодится для редких случаев наследников базового класса диджи.

Название ветки для телескопа формируется как ERQTelescope[Si,Csl]Digi__[X|Y-для односторонних станций, XY|YX - для двухсторонних]_[номер станции данного типа(односторонней или двусторонней)]_[X|Y в случае двусторонней]

Station_name - берется из базы данных: <https://github.com/ExpertRootGroup/er/blob/dev/db/QTelescope/QTelescopeParts.xml>

Фактически является идентификатором конкретной железки и поэтому должно быть уникальным. (На картинке это не так, поправим)

Название ветки для BeamDet формируется как BeamDet[MWPC|ToF]Digi[ToFNb |X MWPCNb | Y MWPCNb].

Фактически сейчас для каждого детектора применяется своя система именования ветки. Это порочно, надо от этого

уходить.

Предлагаю ввести название ветки по формату: [Detector_type]_[Detector_name]_[Station_name]_[адресация данного детектора]

Пример Telescope: Telescope_LeftTelescope_DoubleSI2_1_XY_X (Detector_name = LeftTelescope; StationName=DoubleSI2; адресация данного детектора = 1_XY_X (Станция первая в телескопе, к мишени смотрит сторона, восстанавливающая X, в данной ветке лежат диджи, восстанавливающие X))

Пример BeamDet: BeamDet_AculBeamDet_MWPC1_1_X (Detector_name=AculBeamDet; Station_name=MWPC1, 1_X - первая станция по пучку, восстанавливает X)

Так что не надо ориентироваться на те названия веток, что сейчас выдают диджитизаторы. Давайте кашерные названия веток сформируем в рамках данной задачи, остальное подстроим потом.

Также предлагаю начать пользоваться системой Folders FairRoot, чтобы ввести группировки веток для удобства дальнейшего использования. Например:

```
Detector_type
  Detector_name
    Station_name
      ...
```

Это поможет алгоритмам реконструкции проще искать группы веток с которыми необходимо совершать операции. Station_name должна быть уникальной и браться из xml базы данных железок.

Так как, что именно пишется со станции: только время, амплитуда и время или что то еще... Информация об этом должна быть представлена в setup.xml файле.

В дальнейшем мы научим DigiBuilder сохранять эту информацию в FairRunTimeDB с помощью классов типа QTelescopeSetup, BeamDetSetup и т.д, которые ответственны предоставлять классам реконструкции интерфейсы к данной информации. НО не в рамках этой задачи!

Программисткая часть:

ACCULINNA_go4_user_library должна уметь инсталлиться:

./compile.sh должен принимать как параметр install_prefix

Лучше, если процедура сборки будет оформлена с помощью stake. sh скрипты - не тру.

И после make install или типо того, в указанной install директории должны создаваться директории:

```
includes/
lib/
```

stake процедура ег научится работь с флагом -DACCULINNA_go4_user_library='install директория'

по которому будет создавать переменные:

```
ACCULINNA_go4_user_library_INCLUDES
```

```
ACCULINNA_go4_user_library_LIBS
```

Добавлять их в config.sh

И только в случае наличия этого флага компилить DigiBuilder.

Класс DigiBuilder должен находиться в директории beamtime.

History

#1 - 07/12/2018 12:30 PM - Vitaliy Schetinin

- Description updated

#2 - 07/16/2018 01:19 PM - Vitaliy Schetinin

- Description updated

#3 - 07/17/2018 12:57 AM - Sergey Belogurov

Единственное замечание, в паре Detecor type, Detector name надо позаботиться о длине названий веток, чтоб их было легко анализировать.

Поэтому Telescope_LeftTelescope... (Detector_type=Telescope, Detector_name = LeftTelescope) кажется избыточно длинным.

Как-то надо покороче Tlscr_LeftT, например.

#4 - 08/20/2018 07:06 AM - Vitaliy Schetin

- Assignee changed from Egor Ovcharenko to Vitaliy Schetin

#5 - 09/13/2018 03:22 PM - Ivan Muzalevsky

Просьбы

- 1) В макрос digibuilder.C(NEvents) добавить возможность обработки нескольких файлов, схожих по имени, либо сразу целой папки с файлами.
Сделать так, чтобы, если NEvents не задан, то обрабатываются все события input файлов.
- 2) Добавить отдельную задачу, где мы будем общаться о том, какие отборы сейчас применяются в digibuilder и в reso.
- 3) Добавить кластеризацию MWPC (вычислять координату в MWPC как среднее арифметическое номеров зажжённых проволочек, если это проволочки рядом)

#6 - 09/17/2018 02:41 PM - Ivan Muzalevsky

- 1) MWPCtimes: временные данные в ветках, относящихся к MWPC (например BeamDetMWPCDigiX1.fTime), не переводятся в наносекунды. Нужно умножить значение времени на 0.125
- 2) ERQTelescopeSiDigi_Left_telescope_SingleSi_SQ300_X_0.fTime не калибруются, нужно умножить значение на 0.3
- 3) ERQTelescopeCslDigi_Left_telescope_Csl_L_0.fTime пишутся только нули.
- 4) ERQTelescopeCslDigi_Right_telescope_Csl_R_0.fTime пишутся только нули
- 5) На данный момент, после конвертации lmd->root с помощью последней версии егоровской библиотеки в ветку DetEventCommon.trigger пишется значение триггера. Нужно, чтобы ER тоже записывал это значение.

В таске ERBeamDetUnpack.cxx строки 129-134 допускают, что в digi.root будут события, где не записалось время MWPC, а только амплитуды. Нам такие события не нужны, их можно выбрасывать.

#7 - 09/18/2018 08:13 AM - Vitaliy Schetin

- 1) Добавил значение триггера в выходной файл digibuilder.
<https://github.com/ExpertRootGroup/er/commit/04f520cd04dc58d6dcdad34bbe0332799fb7912f>

См. ветку EventHeader.fTrigger

Обрати внимание на необходимость добавить в макрос строки:

```
ERBeamTimeEventHeader* header = new ERBeamTimeEventHeader();  
run->SetEventHeader(header);
```

2) Добавил временные калибровки и в MWPC и в телескопы.

<https://github.com/ExpertRootGroup/er/commit/6329d3f51367a9db5399087d08ae5bae91ead218>

3) Поправил все логи. Теперь ничего не пишется во время обработки. Только если включить DEBUG уровень логов.

<https://github.com/ExpertRootGroup/er/commit/9c997397ba232b917c3e12ce852dee6980e68ef1>

#8 - 09/19/2018 09:20 AM - Vitaliy Schetinin

По поводу отборов. Добавил возможность UserCut в Digibilder и расширил в ERRunAna.

Что это значит? В er можно процессить не все события во входном файле. А только те, которые прошли Cut.

В случае DigiBuilder в макросе это выглядит так:

```
builder->SetUserCut("Beam_detector_F3.@fDetMessages.GetEntriesFast() == 4");
```

Строка в скобках? на самом деле? полноценный рутковский объект типа TCut. То есть туда прямо в макросе можно добавить и Graphical Cut и написать, в принципе, любое выражение, которое мы пишем в tree->Draw("",cut)

Предвдушая вопрос про @: это тоже нативный рутковский синтаксис, позволяющий обратиться к объекту коллекции в дереве. Так что здесь я сразу отфильтровал события, в которых почему то амплитуд в F3 меньше, чем 4. Таких кстати много

Таким образом можно длеать отборы до digibuilder прямо на сырых данных.

Можно делать отборы после дигибилдер в макросе реконструкции, к примеру так:

```
run->SetUserCut("EventHeader.fTrigger==1");
```

Пользуйтест, пожалуйста.

#9 - 09/19/2018 09:23 AM - Vitaliy Schetinin

Забыл. Чтобы заработало, надо подождать пока Егор примет https://github.com/evovch/ACCULINNA_go4_user_library/pull/19

Обновить AccDaq master, обновить er 210_digibuilder

#10 - 09/19/2018 03:00 PM - Ivan Muzalevsky

Сделать так, чтобы при запуске digibuilder, гесо в терминал выводился прогресс обработки. Каждые 10000 событий вывод о том, сколько событий обработано из сколько, либо какой процент всех событий обработан.

#11 - 09/24/2018 11:29 AM - Ivan Muzalevsky

Необходимо добавить функционал распараллеливания макросов digibuilder и geco на несколько потоков

Для нормальной обработки данных необходимо делать отборы по событиям. на данный момент этот функционал крайне неудобен. Моё предложение - создать несколько веток, в которые можно будет записывать некие индикаторы, флаги, для разных отборов. То есть при выполнении выбранного условия, значению соответствующего флага присваивать TRUE. Флагов должно быть достаточно много

#12 - 09/24/2018 12:08 PM - Ivan Muzalevsky

нужно сделать так, чтобы Cut'ы не резали количество событий, а чтобы для событий которые не прошли отбор, TClonnesArray не заполнялись.
То есть если события не удовлетворяет хоть одному условию, то во всех ветках, размер TClonnesArray равен 0.

#13 - 11/20/2018 03:14 PM - Ivan Muzalevsky

На данный момент, в методе Bool_t ERTelescopeUnpack::ApplyCalibration калибровка значений времени делается следующим образом: <https://github.com/ExpertRootGroup/er/blob/b491ec622a863692eaa615292831c91524bcd754/beamtime/unpack/ERTelescopeUnpack.cxx#L317>

Необходимо изменить метод калибровки времён. Нужно сделать что-то аналогичное амплитудам. В макросе должен сообщаться путь к файлу с калибровочными параметрами. В файле будет находиться два столбца с калибровочными параметрами (par1 par2). Параметры используются для линейной калибровки следующим образом:
 $Time(ns) = time(channel) * par1 + par2$

Для всех кремниевых станций, как и для амплитуд, так и для времён будут свои калибровочные параметры.

#14 - 11/20/2018 05:44 PM - Ivan Muzalevsky

- File err.txt added
- File out.txt added
- File digi.C added

к сожалению у меня не работает digibuilder.C для новых файлов и нового xml-file.

встречаюсь с такой ошибкой:
[FATAL] [20.11.2018 16:59:24] [ERTelescopeUnpack.cxx:CheckSetup:359] Amplitude station DSDX_C not found in setup configuration file!
[FATAL] [20.11.2018 16:59:24] We stop the execution of the process at this point.
[FATAL] [20.11.2018 16:59:24] For later analysis we write a core dump to core_dump_22128

отконвертированный root-файл можно взять тут: a5.jinr.ru/store/EXP201811/he8_05_0002.lmd.root
XML-file: a5.jinr.ru/store/EXP201811/myXML.xml

Скidaю логи и сегг digibuilder и сам макрос digibuilder. В архиве лежат все параметры, которые нужно использовать для калибровок. В новом сегале изменилась геометрия детекторов, добавился новый телескоп: центральный, также изменены правый и левый.

???????? ????????

Beam_detector

- [REDACTED]
- [REDACTED]
- [REDACTED]

Right_telescope

- [REDACTED]
- [REDACTED]
- [REDACTED]

Left_telescope

- [REDACTED]
- [REDACTED]
- [REDACTED]

Central_telescope

- [REDACTED]
- [REDACTED]

#15 - 11/20/2018 05:47 PM - Ivan Muzalevsky

- File *parameters.tar.bz2* added

#16 - 11/22/2018 08:34 PM - Vitaliy Schetinin

- File *digi.C* added

- File *parameters.zip* added

По последнему сообщению Ивана:

Проблема была в опечатках в макросе и недостаточно информативном сообщением об ошибке для того, чтобы это выявить.

Поправленный макрос и отформатированные файлы калибровок прикладываю.

Сообщение об ошибке поправлено: <https://github.com/ExpertRootGroup/er/commit/b36a36d5d85c9e4224b68cd45bb890167deae16>

#17 - 11/23/2018 11:18 AM - Ivan Muzalevsky

Привет! Обнаружилось небольшое неудобство метода `ERTelescopeUnpack::ReadCalFile`

При данном алгоритме считывания столбцов в элементы объекта `TMatrixD` вижу следующее: если в текстовом файле второй столбец не заканчивается ровно на последней цифре числа, а есть ещё какой нибудь отступ от него, то возникает ошибка со следующим сообщением:

```
[FATAL ] [23.11.2018 11:11:49] [ERTelescopeUnpack.cxx::ReadCalFile:300] Wrong file format in
```

Это терпимо, но причиняет небольшую боль, так как файлы приходится ровнять вручную.

#18 - 12/03/2018 08:13 AM - Vitaliy Schetinin

Ivan Muzalevsky wrote:

На данный момент, в методе Bool_t ERTelescopeUnpack::ApplyCalibration калибровка значений времени делается следующим образом: <https://github.com/ExpertRootGroup/er/blob/b491ec622a863692eaa615292831c91524bcd754/beamtime/unpack/ERTelescopeUnpack.cxx#L317>

Необходимо изменить метод калибровки времён. Нужно сделать что-то аналогичное амплитудам. В макросе должен сообщаться путь к файлу с калибровочными параметрами. В файле будет находиться два столбца с калибровочными параметрами (par1 par2). Параметры используются для линейной калибровки следующим образом:
 $Time(ns) = time(channel) * par1 + par2$

Для всех кремниевых станций, как и для амплитуд, так и для времён будут свои калибровочные параметры.

Сделано

#19 - 12/03/2018 08:20 AM - Vitaliy Schetinin

Иван, давай скомпилируем текущий список доработок в digibuilder:

На данный момент из этой переписки я вижу следующие:

1. Сделать так, чтобы при запуске digibuilder, гесо в терминал выводился прогресс обработки. Каждые 10000 событий вывод о том, сколько событий обработано из скольки, либо какой процент всех событий обработан.
2. Необходимо добавить функционал распараллеливания макросов digibuilder и гесо на несколько потоков
3. Набор претензий к тому, как устроены отборы.

Два вопроса:

1. Это весь список?
2. В чем конкретно претензии к отборам? Желательно на конкретном use case.

#20 - 12/09/2018 10:45 PM - Ivan Muzalevsky

При использовании UserCut выяснилось следующее:

Если данное событие не прошло условия отбора, то хотя размер всех TClonesArray оказывается равен 0, значение триггера тем не менее заполняется значениями, предположительно равным значению триггера в последнем "нормальном" событии. (событие прошедшее отбор UserCut). Нужно как то пофиксить этот баг, либо не заполнять совсем, либо заполнять бессмысленным значением. Также нужно позволить пользователю самому выбирать опции сохранять количество событий после digibuilder или резать, при использовании UserCut

#21 - 12/10/2018 06:48 AM - Vitaliy Schetinin

Поправил.

1. Когда событие не проходит кат, триггер в его EventHeader заполняется -1:

<https://github.com/ExpertRootGroup/er/commit/f078dbc2610c760b18c3f19dabed8a2e9e68a60c>

2. В методе ERDigibuilder::SetUserCut введен второй параметр - опция fFillSkippedEvents.

По умолчанию она true и событие, которое не проходит кат, попадает в выходной файл с пустыми TClonesArray.

Если опция false:

```
builder->SetUserCut("Beam_detector_F3.@fDetMessages.GetEntriesFast() == 4",kFALSE);
```

Событие в выходной файл не попадет. Будет разное количество событий во входном и выходном файле.

<https://github.com/ExpertRootGroup/er/commit/0e13f326e2fb520868bab0a1a509ca9e19875e74>

#22 - 02/27/2019 07:11 AM - Vitaliy Schetinin

- Status changed from *Открыта* to *Закрыта*

- % Done changed from 0 to 100

Files

digiTTree.png	16.9 KB	07/11/2018	Vitaliy Schetinin
out.txt	1.79 KB	11/20/2018	Ivan Muzalevsky
err.txt	25.3 KB	11/20/2018	Ivan Muzalevsky
dig.i.C	3.52 KB	11/20/2018	Ivan Muzalevsky
parameters.tar.bz2	7.34 KB	11/20/2018	Ivan Muzalevsky
dig.i.C	3.43 KB	11/22/2018	Vitaliy Schetinin
parameters.zip	12.3 KB	11/22/2018	Vitaliy Schetinin