

Développement en C++ de logiciels et de bibliothèques modulaires pour le pilotage, l'acquisition, la calibration et le diagnostic de la caméra médicale xemis 2

Olivier Lemaire

lemaire@subatech.in2p3.fr

CNRS / IN2P3 / SUBATECH

Exposé du **Mercredi 14 juin 2023 à 14h15**

Compilé le : 14 juin 2023

1 PRÉSENTATION DE L'EXPÉRIENCE

■ objectif et principe de fonctionnement

- Objectifs
- Principe

■ chaîne d'électronique

- schéma bloc
- schéma visuel de la chaîne
- la réalisation
- l'avancement

2 ENVIRONNEMENT DE DÉVELOPPEMENT

■ logiciels, bibliothèques, langage

■ besoins et contraintes

■ les logiciels développés

- Les projets et leur dépendances

3 STRATÉGIE DE MUTUALISATION

■ création de bibliothèques

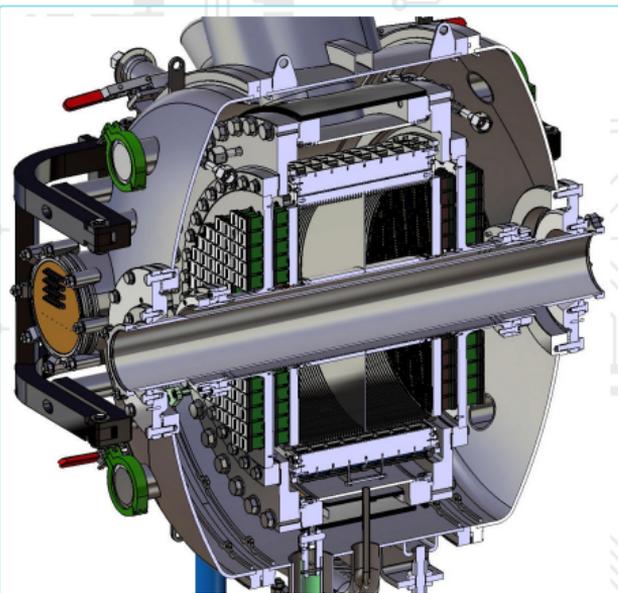
- git
- bibliothèques simples
- bibliothèques hybrides
- bibliothèques externes
- les bibliothèques partagées localement, le cas x2_daq_visu
- la cartographie des relies, le cas tordu ?

4 PRÉSENTATION DE QUELQUES LOGICIELS

■ x2_daq_visu

- spartan
- files
- acquisition
- data display
- plot
- print

5 CONCLUSION



→ Xenon Medical Imaging System

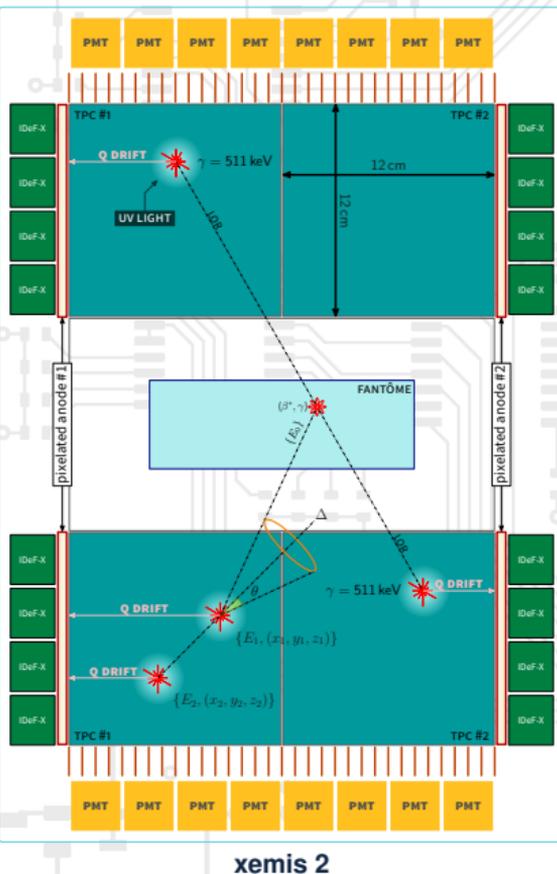
- caméra compton au xénon liquide
- imagerie 3 photons

→ objectif

- produire des images médicales de bonne résolution à partir d'une très faible activité
 - 100 fois moins par rapport au TEP classique

→ technologie

- détecteur chambre à projection temporelle
- milieu de détection xénon liquide
- radio-isotope scandium 44 émetteur 3 photons (β^+ , γ)



→ principe de détection

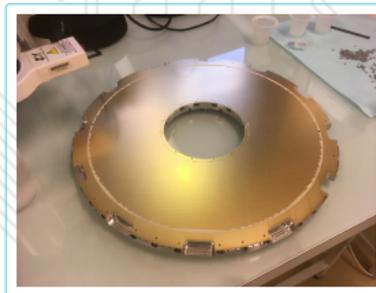
→ 2 photons émis dos à dos

- émission de lumière UV, temps t_0
- dépôts de charge qui dérivent localisation (x, y) et un temps d'arrivée
- $(x, y, \Delta t) \rightarrow (x, y, z)$ car vitesse de dérive connue
- reconstruction d'une ligne de réponse

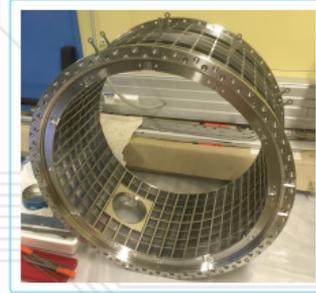
→ un 3^e photon

- même principe, on cherche $\{E_1, (x_1, y_1, z_1)\}$ et $\{E_2, (x_2, y_2, z_2)\}$
- $$\cos(\theta) = 1 - m_e c^2 \frac{E_1}{E_0(E_0 - E_1)}$$

→ en combinant le cone et la ligne de réponse on obtient l'origine de la désintégration



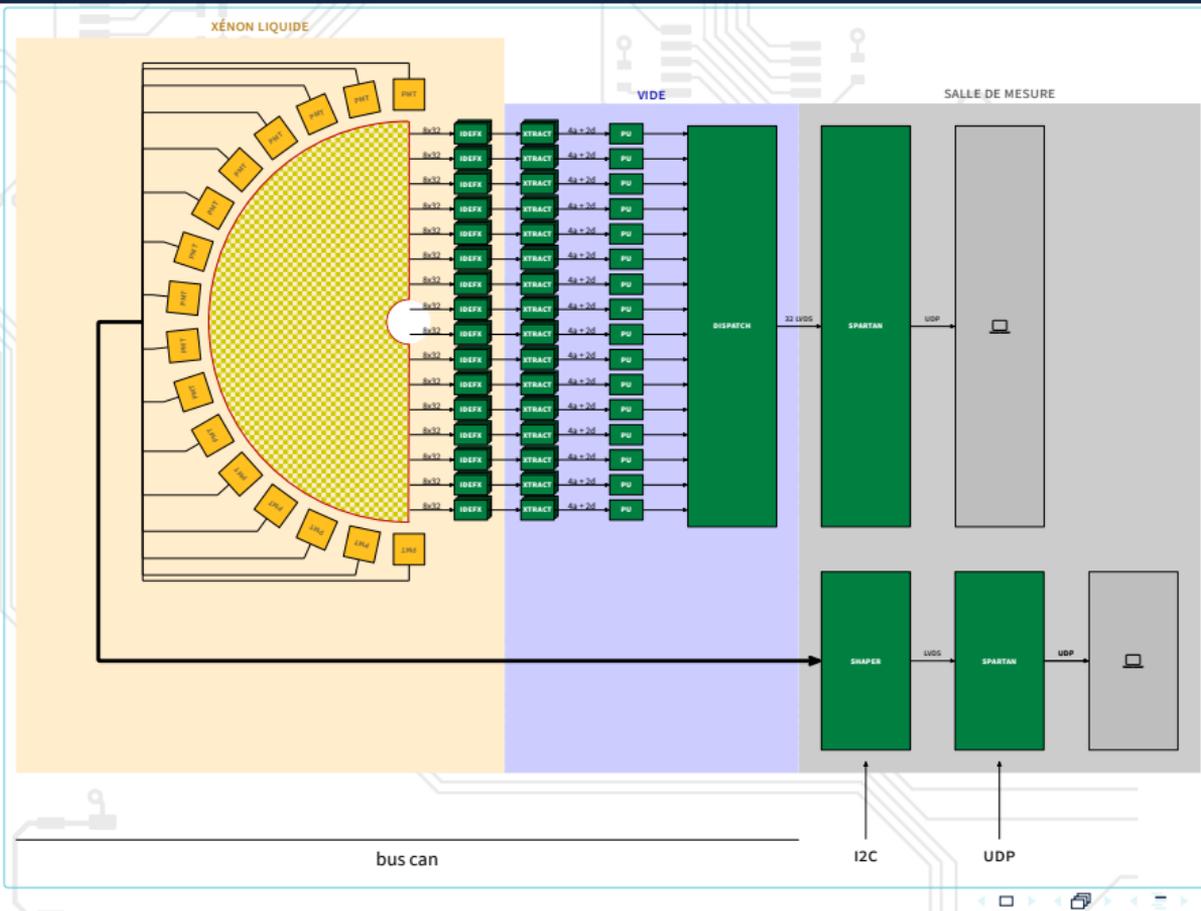
anode

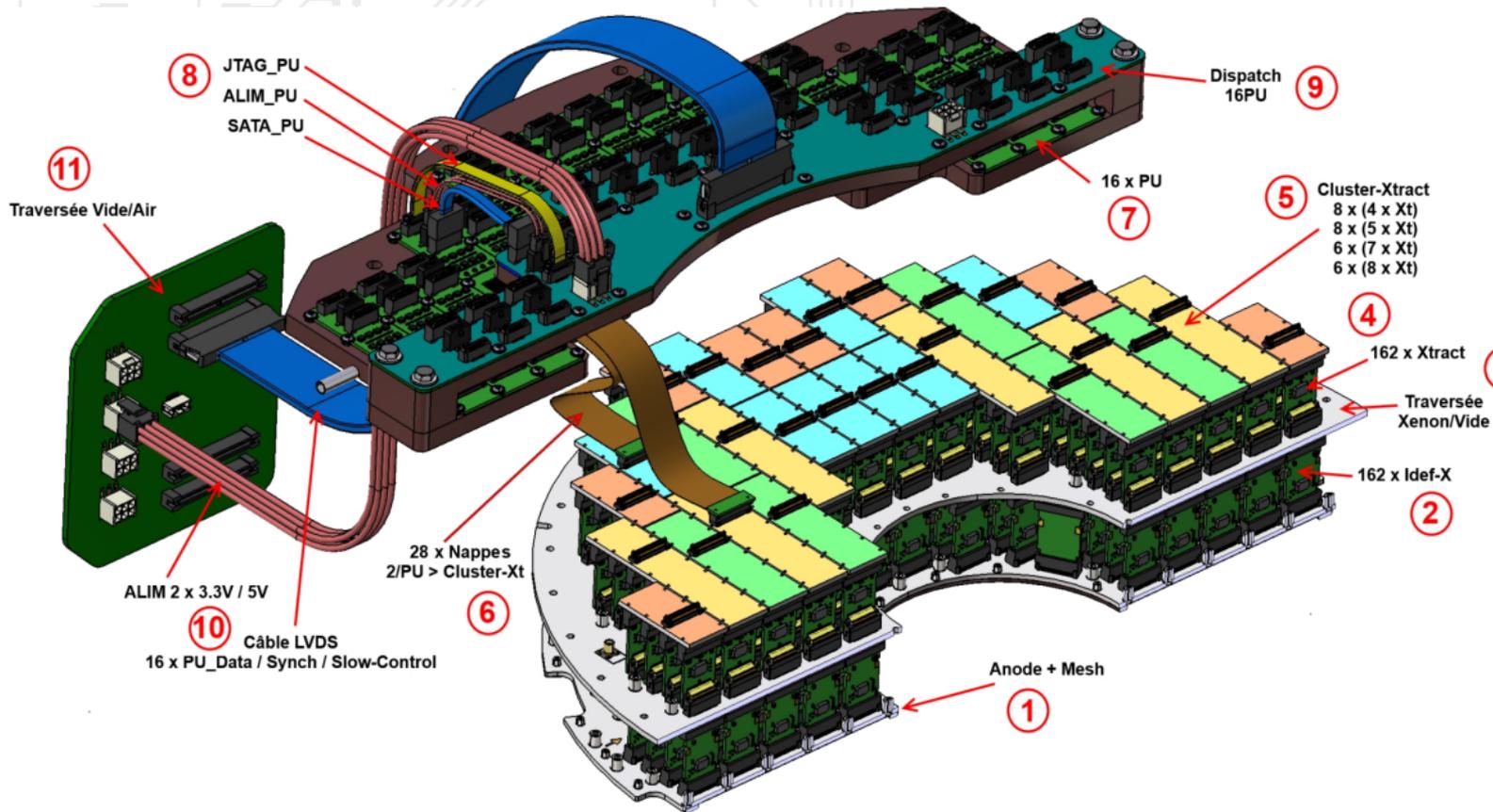


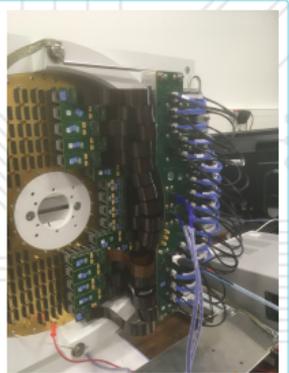
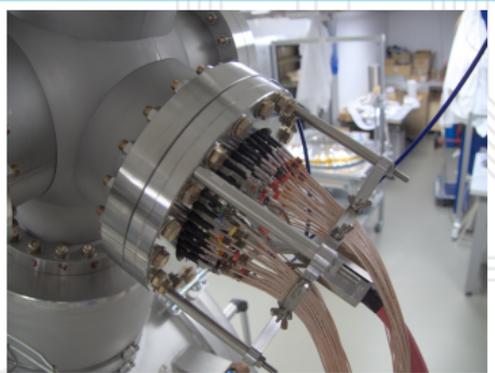
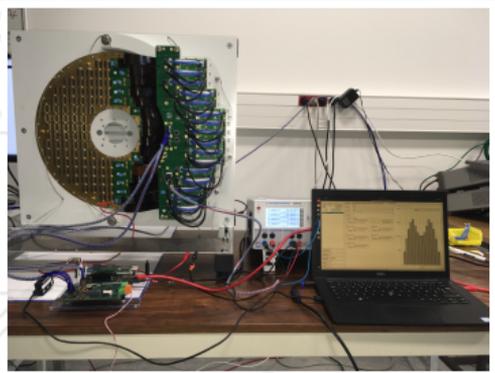
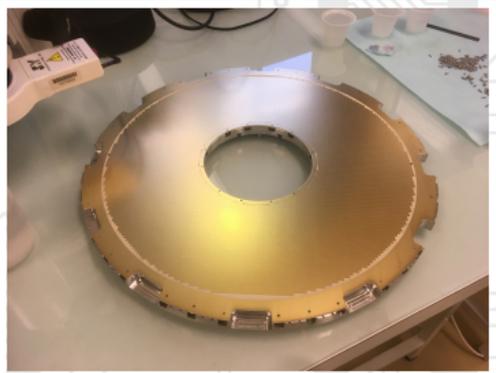
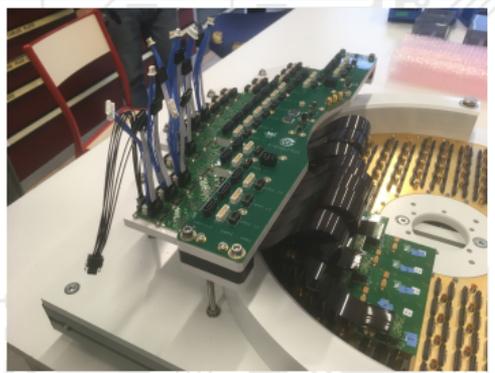
cage pmt



xemis

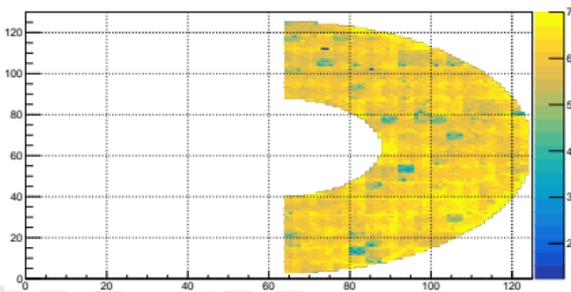








map of pixel noise



- nous avons un emplacement au CHU du Nantes
- Nous en sommes en phase de montage
 - la méca est "terminée"
 - la baie informatique est prête
 - l'électronique PMT est prête
 - l'électronique charge est montée
- pressage et montage d'une anode la semaine dernière

1 PRÉSENTATION DE L'EXPÉRIENCE

■ objectif et principe de fonctionnement

- Objectifs
- Principes

■ chaîne d'électronique

- schéma bloc
- schéma visuel globale
- la réalisation
- l'avancement

2 ENVIRONNEMENT DE DÉVELOPPEMENT

■ logiciels, bibliothèques, langage

■ besoins et contraintes

■ les logiciels développés

- Les projets et leur dépendances

3 STRATÉGIE DE MUTUALISATION

■ création de bibliothèques

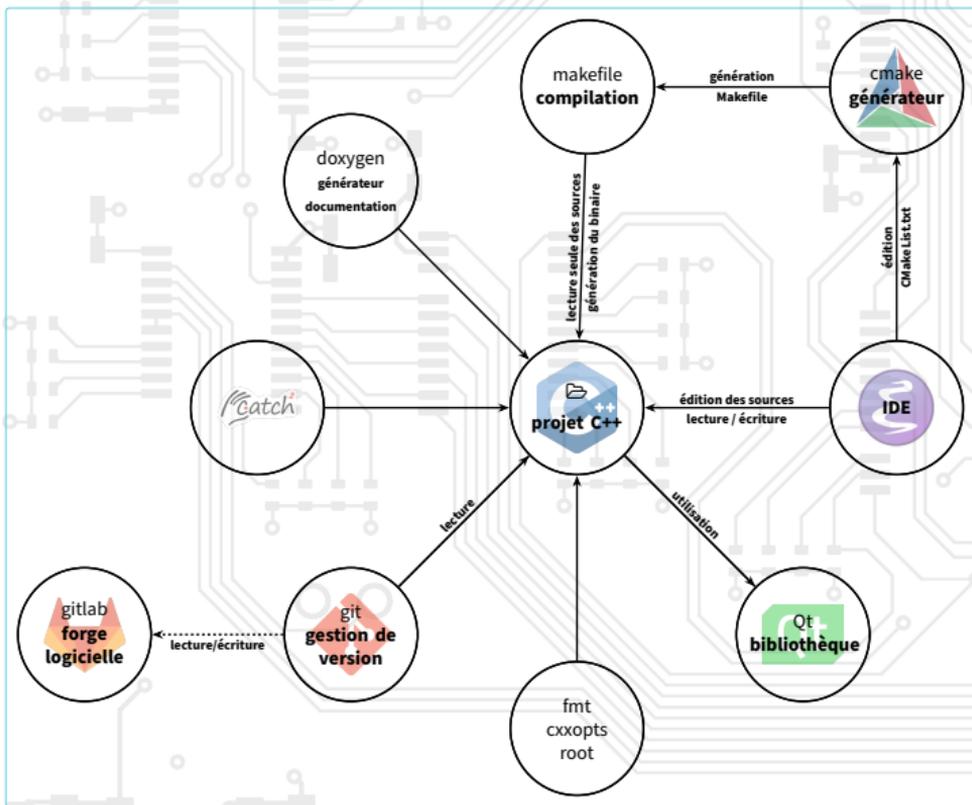
- git
- bibliothèques simples
- bibliothèques hybrides
- bibliothèques externes
- les bibliothèques partagées localement, le cas x2_daq_visu
- la cartographie des relies, le cas tordu ?

4 PRÉSENTATION DE QUELQUES LOGICIELS

■ x2_daq_visu

- spartan
- files
- acquisition
- data display
- plot
- print

5 CONCLUSION



environnement de développement

- langage C++**17**
 - python pour chainer les procédures
- partie graphique Qt**6**
- construction cmake 3.xx
- calcul root
- fmt, cxxopts
 - essayer de limiter les dépendances
- gestion de version git
 - gitlab in2p3 (hébergement)
 - git (backend)
- test catch2
- documentation doxygen
- OS almaLinux (CERN)

besoins et contraintes::

How Projects Really Work (version 1.0)

Create your own cartoon at www.projectcartoon.com

How the customer explained it

How the project leader understood it

How the analyst designed it

How the programmer wrote it

How the business consultant described it



How the project was documented

What operations installed

How the customer was billed

How it was supported

What the customer really needed

→ **les besoins**

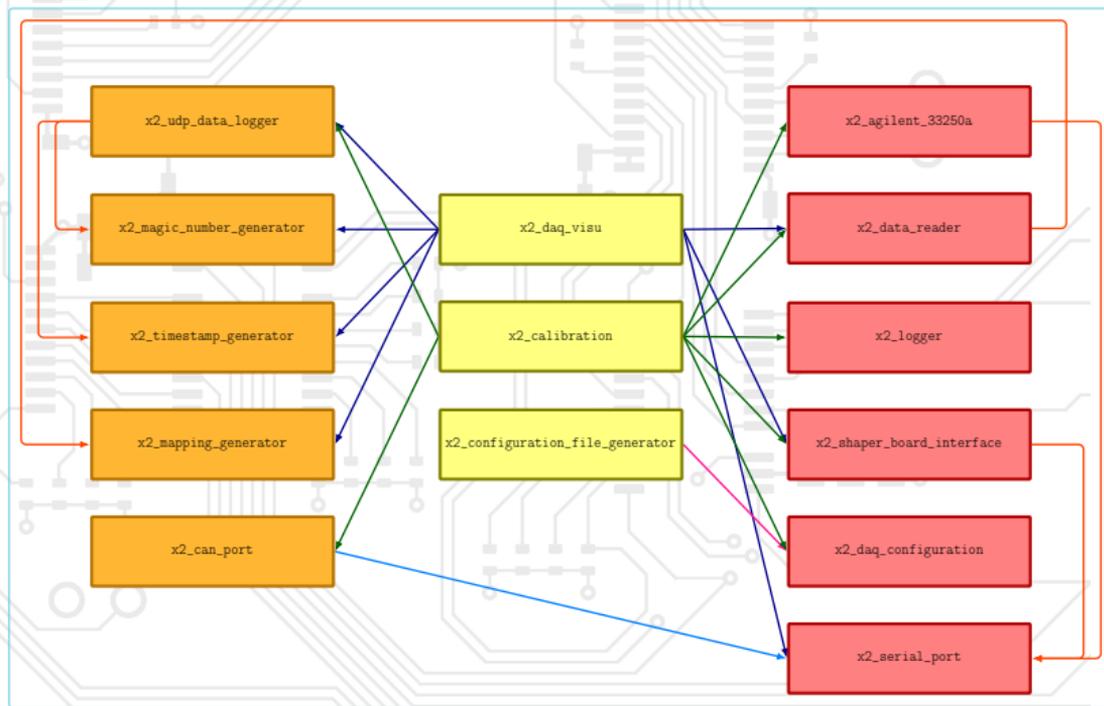
- logiciels de diagnostic
- logiciels de calibration
- logiciels de test
- toujours faire les choses de la même façon

→ **les contraintes**

- développement en parallèle du firmware et du software
 - pas toujours évident de savoir d'où viennent les problèmes
 - beaucoup de choses à développer
 - équipe restreinte
 - des demandes pas toujours faciles

→ **impératif**

- de la reproductibilité
- minimum de maintenance
- transparent pour l'utilisateur (sauf Amaury)



les logiciels développés

- beaucoup de dépendances et de composants mutualisables
- **nécessité de trouver une stratégie...**

1 PRÉSENTATION DE L'EXPÉRIENCE

■ objectif et principe de fonctionnement

- Objectifs
- Principe

■ chaîne d'électronique

- schéma bloc
- schéma visuel globale

- la réalisation
- l'avancement

2 ENVIRONNEMENT DE DÉVELOPPEMENT

■ logiciels, bibliothèques, langage

■ besoins et contraintes

■ les logiciels développés

- Les projets et leur dépendances

3 STRATÉGIE DE MUTUALISATION

■ création de bibliothèques

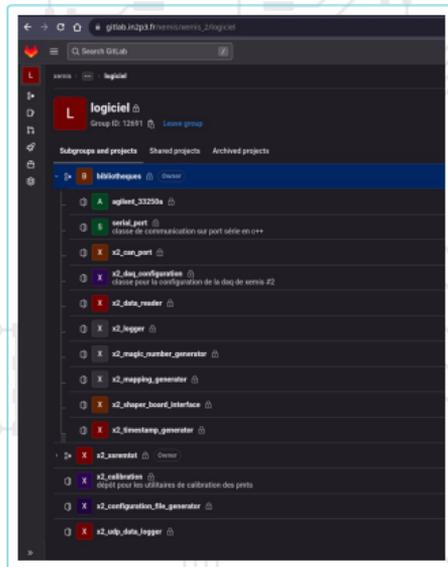
- git
- bibliothèques simples
- bibliothèques hybrides
- bibliothèques externes
- les bibliothèques partagées localement, le cas x2_daq_visu
- la cartographie des relies, le cas tordu ?

4 PRÉSENTATION DE QUELQUES LOGICIELS

■ x2_daq_visu

- spartan
- files
- acquisition
- data display
- plot
- print

5 CONCLUSION



bibliothèques sur le dépôt gitlab



- les logiciels utilisent souvent les mêmes composants
 - nécessité de créer des bibliothèques
 - simplifier la maintenance et assurer de la reproductibilité
- se débarrasser de cpold

```
tree --charset unicode --gitignore
|-- foo_dev
|   |-- foo.vi
|   |-- foo.vi.old
|   |-- foo.vi.old.2009_03_29
|   |-- foo.vi.marche_pas
|   |-- foo.vi.todelete
|   |-- foo.vi.OLD.2006_05_12
|   |-- foo.vi.check
foo_save_dev.rar
foo_save20090329_v2_0.tgz
foo_save20080412_v1_2.tgz
foo_save20060509_v1_0.tgz
```

- git / gitlab
 - tout regrouper au même endroit
 - utilisation des groupes et sous-groupes pour hiérarchiser le travail
 - donne la possibilité d'utiliser des sous-modules
 - propagation des modifications contrôlée
 - convention de nommage, exemple x2_sub_project_name

```

tree --charset unicode --gitignore
.
|-- _clang-format
|-- _clang-tidy
|-- cmake
|-- CMakeLists.txt
|-- Doxyfile
|-- hooks
|-- include
|   |-- daq_acquisition.hxx
|   |-- daq_common.hxx
|   |-- daq_data.hxx
|   |-- daq_field.hxx
|   |-- daq_file.hxx
|   |-- daq_reader_charge.hxx
|   |-- daq_reader_light.hxx
|   |-- daq_reader_slice.hxx
|   `-- daq_slice.hxx
|-- lib
|-- readme.md
|-- src
|   |-- daq_common.cxx
|   |-- daq_file.cxx
|   |-- daq_reader_charge.cxx
|   |-- daq_reader_light.cxx
|   |-- daq_reader_slice.cxx
|   `-- lib
|       `-- x2_mapping_generator
-- test
    |-- daq_file_generator.cxx
    |-- daq_file_generator.hxx
    `-- test_x2_data_reader.cxx

9 directories, 22 files

```

→ toutes les bibliothèques sont organisées de la même façon

→ structure du répertoire

→ cmake	scripts cmake
→ doc	documentation
→ hooks	fichier git
→ include	fichiers header
→ src	fichiers source
→ test	fichiers de tests
→ lib	bibliothèques

→ un fichier CMakeLists.txt primordial dans cette organisation

→ autres fichiers : clang, doxygen, readme ...

→ objectifs :

→ simplifier en ayant une organisation "conventionnelle" et identique dans tous les sous-projets.

→ simplifier l'ajout à un autre projet le plus possible

→ `git submodule add <relative_url> <destination_folder>`

→ `add_subdirectory(/path/to/library)`

→ `target_link_libraries(main_project PRIVATE <target_lib>)`

```

    ### main project
set(X2_FILE_CAN_SENDER_MAIN_PROJECT OFF)
if(CMAKE_CURRENT_SOURCE_DIR STREQUAL CMAKE_SOURCE_DIR)
  set(X2_FILE_CAN_SENDER_MAIN_PROJECT ON)
endif()

# [...]

if(X2_FILE_CAN_SENDER_MAIN_PROJECT)
# executable
add_library(x2_can src/x2_can_port.cxx)
# [récupération automatique du sous-module x2_serial_port]
target_include_directories(x2_can PUBLIC
  "${CMAKE_SOURCE_DIR}/include")
target_link_libraries(x2_can PUBLIC x2_serial_port cxxopts
  fmt::fmt-header-only)
set(CMAKE_INCLUDE_CURRENT_DIR ON)
set(PROJECT_SOURCES main.cxx )
add_executable(x2_file_can_sender ${PROJECT_SOURCES})
# [...]
else()
add_library(x2_can_port src/x2_can_port.cxx )
target_include_directories(x2_can_port PUBLIC
  ${CMAKE_CURRENT_SOURCE_DIR}/include
  ${CMAKE_BINARY_DIR})

# à gérer dans le projet principal
target_link_libraries(x2_can_port PUBLIC
  Threads::Threads
  cxxopts
  fmt::fmt-header-only
  serial_port)
endif()

```

→ projet pouvant être utilisés soit sous en exécutable, soit sous forme de bibliothèques.

→ exemplex2_can_sender

→ bibliothèque CAN

→ exécutable pour envoyer des fichiers sur le bus CAN

→ même structure que le répertoire précédent mais avec un main.cxx en plus à la racine

→ **bibliothèque**

→ `git submodule add ../bibliotheque/x2_can_sender src/lib`

→ `add_subdirectory(src/lib/x2_can_sender)`

→ `target_link_libraries(main_project PRIVATE x2_can_sender)`

→ **projet principal :**

→ `$ git clone --recursive git@gitlab.in2p3.fr:xemis/[...]/x2_can_port.git`

→ `$ cd x2_can_port`

→ `$ mkdir build && cd build`

→ `$ cmake .. && cmake --build .`

```
# -----
# modules
# -----
include(FetchContent)
# -----
# cxxopts
# -----
find_package(cxxopts 3 QUIET)

# téléchargement de la bibliothèque si elle n'est pas disponible
if(NOT cxxopts_FOUND)
  message(STATUS "cxxopts not found; class downloaded from github")
  FetchContent_Declare(
    cxxoptslib
    GIT_REPOSITORY https://github.com/jarro2783/cxxopts.git
    GIT_TAG         c74846a891b3cc3bfa992d588b1295f528d43039 # v3.0.0
    GIT_PROGRESS   ON
  )
  FetchContent_MakeAvailable(cxxoptslib)
  option(CXXOPTS_SYS "Use cxxopts from git" OFF)
else()
  option(CXXOPTS_SYS "Use cxxopts installed on computer" ON)
endif()

configure_file(${CMAKE_SOURCE_DIR}/templates/cxxopts_headers.hxx.in
  cxxopts_headers.hxx)
```

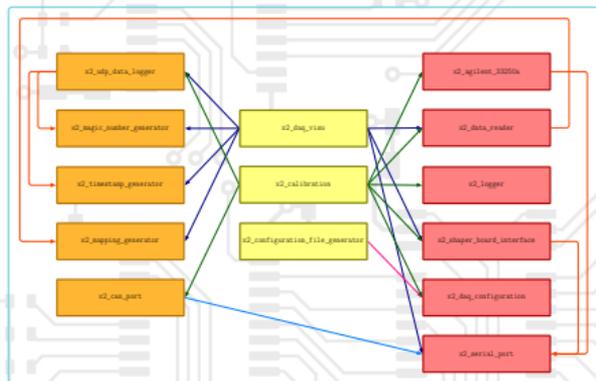
```
#cmakedefine01 CXXOPTS_SYS
// si cxxopts n'est pas installé
#if CXXOPTS_SYS == 0
#include "cxxopts.hpp"
#else
#include <cxxopts.hpp>
#endif
```

```
#define CXXOPTS_SYS 0
// si cxxopts n'est pas installé
#if CXXOPTS_SYS == 0
#include "cxxopts.hpp"
// sinon
#else
#include <cxxopts.hpp>
#endif
```

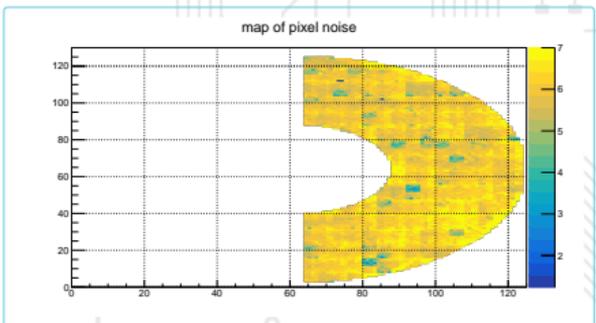
→ récupération de bibliothèques externes

- possibilité de récupération en sous-modules (git)
- téléchargement à l'élaboration (FetchContent)

- 1 vérifier si la paquet existe
- 2 le télécharger dans le cas contraire
- 3 le rendre disponible
- 4 générer un header



les logiciels développés



résultat

- projet regroupant les utilitaires de calibration, test et diagnostic
 - mesure du bruit moyen sur tous les pixels
 - mesure du niveau dc sur tous les pixels
 - calibration du seuil des pmts...
- à chaque procédure correspond un exécutable
- ils partagent tous les même bibliothèques
- certaines bibliothèques sont grosses...
- bibliothèques partagées ou statiques ?
 - statiques → chaque utilitaire embarque sans propre copie
 - partagées → installation ??
- installation d'un rpath sur une bibliothèque partagées
 - le chemin vers les bibliothèques est relatif à l'emplacement de l'utilitaire

création de bibliothèques::les bibliothèques partagées localement, le cas x2_calibration

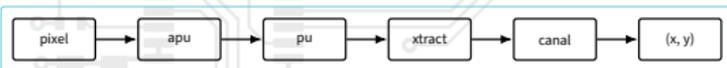
```
# -----
# rpath
# -----
# use, i.e. don't skip the full RPATH for the build tree
set(CMAKE_SKIP_BUILD_RPATH FALSE)
# when building, don't use the install RPATH already (but later on when installing)
set(CMAKE_BUILD_WITH_INSTALL_RPATH FALSE)
set(CMAKE_INSTALL_RPATH "$ORIGIN")
# add the automatically determined parts of the RPATH
set(CMAKE_INSTALL_RPATH_USE_LINK_PATH TRUE)
```

```
build (feature_calibration_zc_th) $$ ldd src/pmt/thresholds/x2_calibration_pmt_thresholds
linux-vdso.so.1 (0x00007ffd5fcd000)
libx2_shaper_board_interface.so => /home/lemaire/code/cxx/xemis/daq/x2_calibration/build/src/lib/x2_shaper_board_interface/libx2_shaper_board_interface.so (0x00007fb03e91000)
libx2_udp_data_logger.so => /home/lemaire/code/cxx/xemis/daq/x2_calibration/build/src/lib/x2_udp_data_logger/libx2_udp_data_logger.so (0x00007fb03e91000)
libx2_data_reader.so => /home/lemaire/code/cxx/xemis/daq/x2_calibration/build/src/lib/x2_data_reader/libx2_data_reader.so (0x00007fb03e1c000)
[...]
```

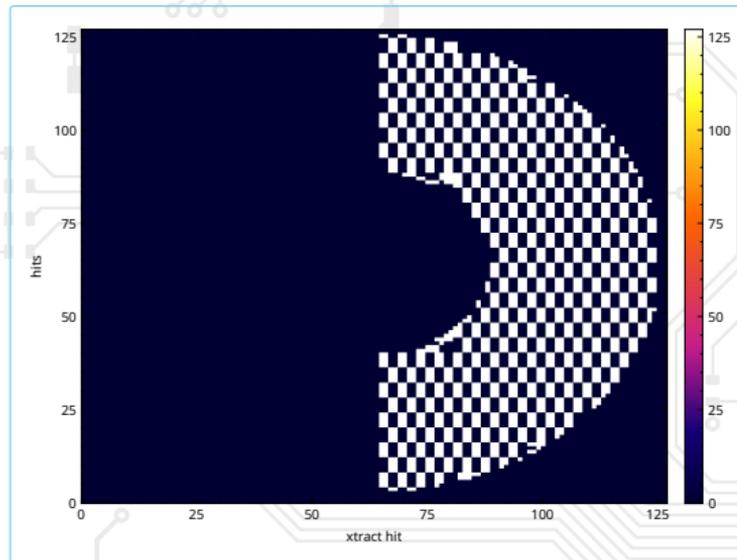
```
cmake --install .
cd ./bin
ldd ./x2_calibration_pmt_thresholds
```

```
total 28M
[...]
-rwxr-xr-x. 1 lemaire lemaire 1.6M May 23 17:44 libx2_data_reader.so
-rwxr-xr-x. 1 lemaire lemaire 246K May 23 17:44 libx2_shaper_board_interface.so
-rwxr-xr-x. 1 lemaire lemaire 2.6M May 24 15:35 libx2_udp_data_logger.so
-rwxr-xr-x. 1 lemaire lemaire 6.0M May 24 15:35 x2_calibration_pmt_thresholds
[...]
```

```
linux-vdso.so.1 (0x00007fff52da0000)
libx2_shaper_board_interface.so => /home/lemaire/code/cxx/xemis/daq/x2_calibration/build/bin/./libx2_shaper_board_interface.so
libx2_udp_data_logger.so => /home/lemaire/code/cxx/xemis/daq/x2_calibration/build/bin/./libx2_udp_data_logger.so (0x00007fbc0692c000)
libx2_data_reader.so => /home/lemaire/code/cxx/xemis/daq/x2_calibration/build/bin/./libx2_data_reader.so (0x00007fbc068b7000)
[...]
```



chaîne d'acquisition simplifiée



exemple de fichier cartographié

→ projet qui génère la cartographie des voies sous forme d'une table de correspondance (lut) dans un fichiers *.hxx

→ à chaque combinaison (d, a, p, x, c) on fait correspondre un pixel (d, x, y)

→ à chaque (d, x, y) → (d, a, p, x, c)

→ petit projet c++ qui fait ce travail

→ conversion de fichiers csv vers mapping.hxx

→ création d'un dépôt, etc.

→ projet simple dans le principe mais l'erreur est humaine...

→ besoin régénérer les tables à chaque modification et de récupérer le nouveau fichier dans les projets qui l'utilisent

→ nécessité de trouver une solution sans pour autant perdre les travail déjà fait...

```

set(X2_MAPPING_EXTRACTOR OFF)
if(CMAKE_CURRENT_SOURCE_DIR STREQUAL CMAKE_SOURCE_DIR)
  set(X2_MAPPING_EXTRACTOR ON)
endif()
[...]
message(STATUS "building project as a library...")

# compilation du générateur
add_custom_target(x2_mapping_generator ALL
  COMMAND g++ "${CMAKE_CURRENT_SOURCE_DIR}/main.cxx" -o generator
  WORKING_DIRECTORY "${CMAKE_BINARY_DIR}"
  COMMENT "building mapping generator"
  VERBATIM)

# execution du générateur et génération du mapping
add_custom_target(x2_mapping_generation ALL
  COMMAND "${CMAKE_BINARY_DIR}/generator"
  DEPENDS x2_mapping_generator
  WORKING_DIRECTORY "${CMAKE_BINARY_DIR}"
  COMMENT "x2 mapping generation"
  VERBATIM
)

# bibliothèque d'interface
add_library(x2_mapping INTERFACE)
target_include_directories(x2_mapping INTERFACE "${CMAKE_BINARY_DIR}")
add_dependencies(x2_mapping x2_mapping_generation)

```

→ bibliothèque hybride

- ce n'est pas vraiment une bibliothèque
- c'est un exécutable qui génère le fichier dont on fera une bibliothèque
- cmake à la rescousse...

- 1 compiler le générateur
- 2 exécuter le générateur pour générer le fichier mapping.hxx
- 3 créer une bibliothèque avec ce nouveau fichier
- 4 créer une chaîne de dépendance

```

inline uint16_t get_index_dapxc_to_dxy(const unsigned int& dispatch, //
  const unsigned int& apus, //
  const unsigned int& pu, //
  const unsigned int& xtract, //
  const unsigned int& channel) //
{
  auto d = dispatch % 2;
  uint16_t index = (d & 0b1) << 13 | // dispatch 1 bit
    ((apus & 0xF) << 9) | // apus -- 4 bits
    ((pu & 0b1) << 8) | // pu -- 1 bit
    ((xtract & 0b111) << 5) | // xtract -- 3 bits
    ((channel & 0x1F)); // canal -- 5 bits

  return index; };

static const std::array<mapping::coordinate_dxy, 16384> lut_dapxc_to_dxy = {
  {0, 77, 7},{0, 78, 7},{0, 76, 7},{0, 79, 7},{0, 77, 6},{0, 78, 6},{0, 76, 6},
  {0, 79, 6}, [...] };

static const std::array<mapping::coordinate_dapxc, 16384> lut_dxy_to_dapxc =
  { [...] };

```

1 PRÉSENTATION DE L'EXPÉRIENCE

■ objectif et principe de fonctionnement

- Objectifs
- Principe

■ chaîne d'électronique

- schéma bloc
- schéma visuel de la chaîne
- la réalisation
- l'avancement

2 ENVIRONNEMENT DE DÉVELOPPEMENT

■ logiciels, bibliothèques, langage

■ besoins et contraintes

■ les logiciels développés

- Les projets et leur dépendances

3 STRATÉGIE DE MUTUALISATION

■ création de bibliothèques

- git
- bibliothèques simples
- bibliothèques hybrides
- bibliothèques externes
- les bibliothèques partagées localement, le cas x2_calibration
- la cartographie des relies, le cas tordu ?

4 PRÉSENTATION DE QUELQUES LOGICIELS

■ x2_daq_visu

- spartan
- files
- acquisition
- data display
- plot
- print

5 CONCLUSION

daq file forensic software -- git commit: 04ab90d34f3c86a158d884eca93b366600549046, git branch: dev

spartan configuration files acquisition raw data display plot textual slices summary pmts

action

charge configuration

configure register(s)
check all registers
uncheck all registers

charge generator configuration

mode: ready

channel: 0

extract: 0

bit: 0

packet: 0

time: 0

value: 0

delay between packets: 0x0

charge input enable

block: 0 - 0

read: 0x00000000 read: 0x00000000

charge channel enable

block: 0 - 0

read: 0x00000000 read: 0x00000000

leds

mode: value

polarity: high

color: green

value: 10

timer: 1000

register generic

block: 000

register: 000

value hex: 0x-00000000

light configuration

light generator configuration

mode: ready

time up: 0

time down: 0

light input enable

block: 0 - 0

read: 0x00000000 read: 0x00000000

light input inverted

block: 0 - 0

read: 0x00000000 read: 0x00000000

light channel enable

block: 0 - 0

read: 0x00000000 read: 0x00000000

acquisition

time high: 10 ms

time low: 80 ms

number of slices: 10

ready to go !!

daq file forensic software -- git commit: 04ab90d34f3c86a158d884eca93b366600549046, git branch: dev

spartan configuration files acquisition raw data display plot textual slices summary pmts

folder /home/lemaire/documents/data/x2_20230414_105031/x2_20230414_105031 open...

add remove file header

Name	Size
.	
..	
x2_20230414_105031_#0.x2daq	428.27 KiB

add to list remove from list clear file list

1 /home/lemaire/documents/data/x2_20230414_105031/x2_20230414_105031/x2_20230414_105031_#0.x2daq

ready to go !!

load files read file header read acquisition header

file Displayed me/lemaire/documents/data/x2_20230414_105031/x2_20230414_105031/x2_20230414_105031_#0.x2daq

file header	acquisition header
file header version (hex) 1	header version 8
machine id (hex) 1ce06475e8c8635d6c33d878e	size 5
magic number (hex) 7359895620264541569	pu in acquisition 0
ms since epoch (hex) 66239a127dd23d81	25 MHz 0
date and time	lx150 1
reset daq time low (dec) 80	pu number 18
reset daq time high (dec) 10	delay between slices 296020
number of slices (dec) 10	firmware version 2
git hash (hex) 7c3bd066e896510f7be5c7346	pm in acquisition
file number (hex) 0	slave 1
	pm number 4
	pu address length fifo dbu 0
	pm address length fifo dbu 0
	udp max length 0
	build date and time XXXXXXXXXXX
	slice number 3456517982
	slice duration 2357605846

daq file forensic software -- git commit: 04ab90d34f3c86a158d884eca93b366600549046, git branch: dev

spartan configuration files acquisition raw data display plot textual slices summary pmts

time base

time high 10

time low 80

number of slices 10

save path /home/lemaire/documents/data/test_banc_lumiere

acquisition mode master

single acquisition

multi acquisition

can configuration

serial port open can port

can baudrate [kbps] close can port

mode get firmware version

number of frame per burst 15 frames get hardware version

delay between frames (ms) 20 ms rescan ports

delay between bursts (ms) 20 ms get serial #

delay end of file (ms) 50 ms send file over can...

restart dingle between bursts don't restart can port closed

firmware version

hardware version

serial number

ready to go !!

folder /home/lemaire/documents/data/x2_20230414_105031/x2_20230414_105031 open...

add to list remove from list clear file list

file name

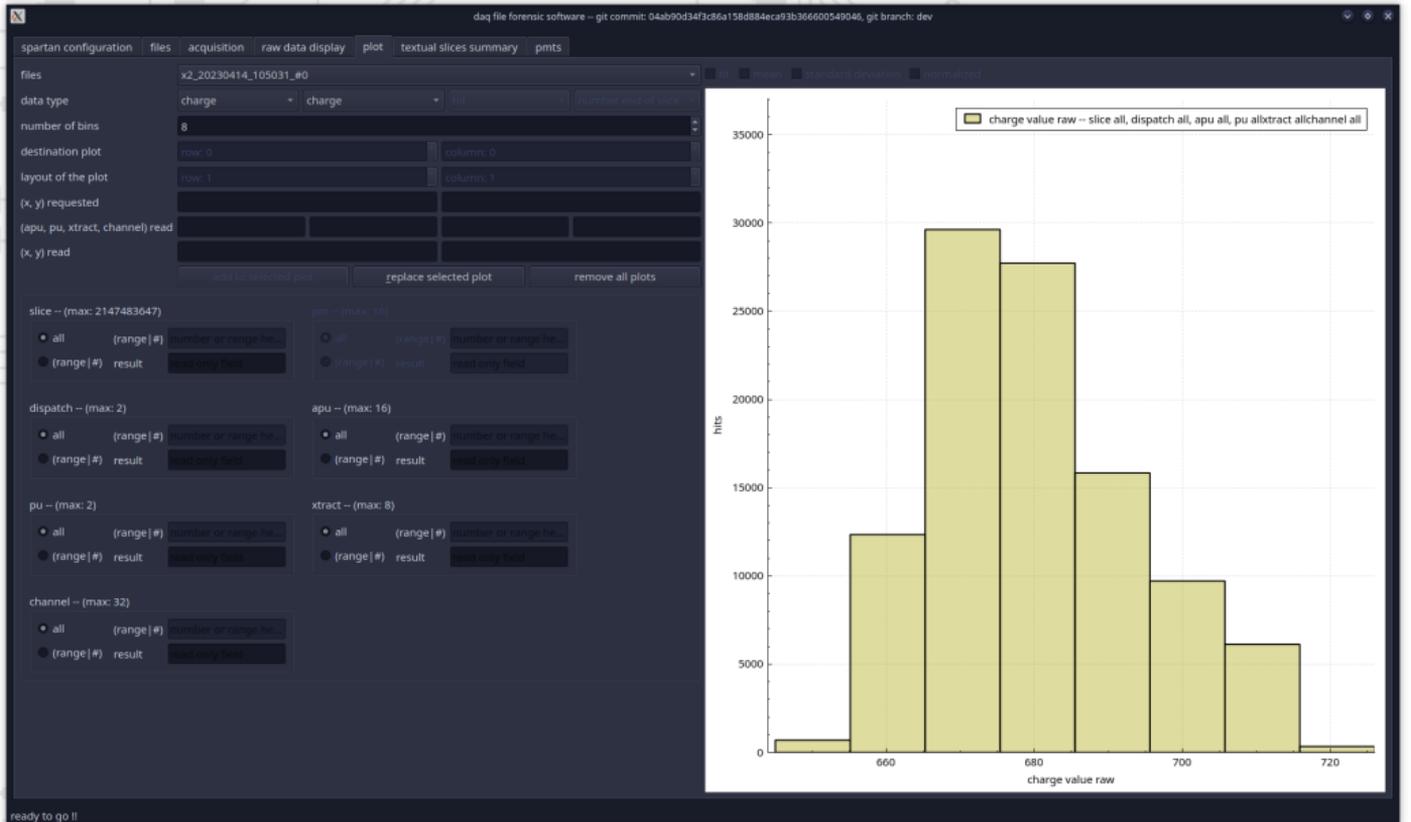
x2_daq_visu::raw data display

daq file forensic software - git commit: 04ab90d34f3c66115b6884eca93b366600549046, git branch: dev

spartan configuration files acquisition raw data display plot textual slices summary pmts

slice 0 x2_20230414_105031_#0.x2daq

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
01555C8020	0157800A43	0158292498	0158528230	0158632268	0158729F2C1	0158820A1284	01589128F2D0	0158992C28B3	0159082C3D						
015912F23018	01592611224	0159329F0388	0159426278	015951842C3	01595A222C3	01596A2028A	0159708A2247	0159792A2247	0159882A2247						
015122A2246	0151329E240	0151429C240	015152A9253	015162902A7	015172A0287	015182A8298	015192A42B3	0151A2962A8	0151B205286						
0151C2A22A8	0151D20F240	0151E2A1281	0151F2A9285	0152028C016	015212806A3	015222992C8	015232992C8	015242992C8	015252992C8						
015262928B	015272952C2	015282982A4	015292A42C1	0152A2B27C8	0152B2992C8	0152C2A2288	0152D2962C5	0152E29A28F	0152F2962C8						
0153029128F	015312942A4	015322942A2	015332942A2	015342972A1	0153529A2A2	0153629A2A2	0153729A2A2	0153829A2A2	0153929A2A2						
0153A2A2A0	0153B2A2A0	0153C2A2A0	0153D2A2A0	0153E2A2A0	0153F2A2A0	015402A2A0	015412A2A0	015422A2A0	015432A2A0						
015442220A	0154522272B	01546222C08	01547222E08	01548222F20	01549223020	0154A223220	0154B223420	0154C223620	0154D223820						
0154E223A20	0154F223D24	01550223F24	01551224230	01552224520	01553224820	01554225220	01555225520	01556225820	01557226220						
01558226520	01559226820	0155A226B20	0155B226E20	0155C227120	0155D227420	0155E227720	0155F228220	01560228520	01561228820						
01562229220	01563229520	01564229820	01565230220	01566230520	01567230820	01568231220	01569231520	0156A231820	0156B232220						
0156C232520	0156D232820	0156E233220	0156F233520	01570233820	01571234220	01572234520	01573234820	01574235220	01575235520						
01576235820	01577236220	01578236520	01579236820	0157A237220	0157B237520	0157C237820	0157D238220	0157E238520	0157F238820						
01580239220	01581239520	01582239820	01583240220	01584240520	01585240820	01586241220	01587241520	01588241820	01589242220						
0158A242520	0158B242820	0158C243220	0158D243520	0158E243820	0158F244220	01590244520	01591244820	01592245220	01593245520						
01594245820	01595246220	01596246520	01597246820	01598247220	01599247520	0159A247820	0159B248220	0159C248520	0159D248820						
0159E249220	0159F249520	015A0249820	015A1250220	015A2250520	015A3250820	015A4251220	015A5251520	015A6251820	015A7252220						
015A8252520	015A9252820	015AA253220	015AB253520	015AC253820	015AD254220	015AE254520	015AF254820	015B0255220	015B1255520						
015B2255820	015B3256220	015B4256520	015B5256820	015B6257220	015B7257520	015B8257820	015B9258220	015BA258520	015BB258820						
015BC259220	015BD259520	015BE259820	015BF260220	015C0260520	015C1260820	015C2261220	015C3261520	015C4261820	015C5262220						
015C6262520	015C7262820	015C8263220	015C9263520	015CA263820	015CB264220	015CC264520	015CD264820	015CE265220	015CF265520						
015D0265820	015D1266220	015D2266520	015D3266820	015D4267220	015D5267520	015D6267820	015D7268220	015D8268520	015D9268820						
015DA269220	015DB269520	015DC269820	015DD270220	015DE270520	015DF270820	015E0271220	015E1271520	015E2271820	015E3272220						
015E4272520	015E5272820	015E6273220	015E7273520	015E8273820	015E9274220	015EA274520	015EB274820	015EC275220	015ED275520						
015EE275820	015EF276220	015F0276520	015F1276820	015F2277220	015F3277520	015F4277820	015F5278220	015F6278520	015F7278820						
015F8279220	015F9279520	015FA279820	015FB280220	015FC280520	015FD280820	015FE281220	015FF281520	01600281820	01601282220						
01602282520	01603282820	01604283220	01605283520	01606283820	01607284220	01608284520	01609284820	0160A285220	0160B285520						
0160C285820	0160D286220	0160E286520	0160F286820	01610287220	01611287520	01612287820	01613288220	01614288520	01615288820						
01616289220	01617289520	01618289820	01619290220	0161A290520	0161B290820	0161C291220	0161D291520	0161E291820	0161F292220						
01620292520	01621292820	01622293220	01623293520	01624293820	01625294220	01626294520	01627294820	01628295220	01629295520						
0162A295820	0162B296220	0162C296520	0162D296820	0162E297220	0162F297520	01630297820	01631298220	01632298520	01633298820						
01634299220	01635299520	01636299820	01637300220	01638300520	01639300820	0163A301220	0163B301520	0163C301820	0163D302220						
0163E302520	0163F302820	01640303220	01641303520	01642303820	01643304220	01644304520	01645304820	01646305220	01647305520						
01648305820	01649306220	0164A306520	0164B306820	0164C307220	0164D307520	0164E307820	0164F308220	01650308520	01651308820						
01652309220	01653309520	01654309820	01655310220	01656310520	01657310820	01658311220	01659311520	0165A311820	0165B312220						
0165C312520	0165D312820	0165E313220	0165F313520	01660313820	01661314220	01662314520	01663314820	01664315220	01665315520						
01666315820	01667316220	01668316520	01669316820	0166A317220	0166B317520	0166C317820	0166D318220	0166E318520	0166F318820						
01670319220	01671319520	01672319820	01673320220	01674320520	01675320820	01676321220	01677321520	01678321820	01679322220						
0167A322520	0167B322820	0167C323220	0167D323520	0167E323820	0167F324220	01680324520	01681324820	01682325220	01683325520						
01684325820	01685326220	01686326520	01687326820	01688327220	01689327520	0168A327820	0168B328220	0168C328520	0168D328820						
0168E329220	0168F329520	01690329820	01691330220	01692330520	01693330820	01694331220	01695331520	01696331820	01697332220						
01698332520	01699332820	0169A333220	0169B333520	0169C333820	0169D334220	0169E334520	0169F334820	0169A335220	0169B335520						
0169C335820	0169D336220	0169E336520	0169F336820	0169A337220	0169B337520	0169C337820	0169D338220	0169E338520	0169F338820						
0169A339220	0169B339520	0169C339820	0169D340220	0169E340520	0169F340820	0169A341220	0169B341520	0169C341820	0169D342220						
0169E342520	0169F342820	0169A343220	0169B343520	0169C343820	0169D344220	0169E344520	0169F344820	0169A345220	0169B345520						
0169C345820	0169D346220	0169E346520	0169F346820	0169A347220	0169B347520	0169C347820	0169D348220	0169E348520	0169F348820						
0169A349220	0169B349520	0169C349820	0169D350220	0169E350520	0169F350820	0169A351220	0169B351520	0169C351820	0169D352220						
0169E352520	0169F352820	0169A353220	0169B353520	0169C353820	0169D354220	0169E354520	0169F354820	0169A355220	0169B355520						
0169C355820	0169D356220	0169E356520	0169F356820	0169A357220	0169B357520	0169C357820	0169D358220	0169E358520	0169F358820						
0169A359220	0169B359520	0169C359820	0169D360220	0169E360520	0169F360820	0169A361220	0169B361520	0169C361820	0169D362220						
0169E362520	0169F362820	0169A363220	0169B363520	0169C363820	0169D364220	0169E364520	0169F364820	0169A365220	0169B365520						
0169C365820	0169D366220	0169E366520	0169F366820	0169A367220	0169B367520	0169C367820	0169D368220	0169E368520	0169F368820						
0169A369220	0169B369520	0169C369820	0169D370220	0169E370520	0169F370820	0169A371220	0169B371520	0169C371820	0169D372220						
0169E372520	0169F372820	0169A373220	0169B373520	0169C373820	0169D374220	0169E374520	0169F374820	0169A375220	0169B375520						
0169C375820	0169D376220	0169E376520	0169F376820	0169A377220	0169B377520	0169C377820	0169D378220	0169E378520	0169F378820						
0169A379220	0169B379520	0169C379820	0169D380220	0169E380520	0169F380820	0169A381220	0169B381520	0169C381820	0169D382220						
0169E382520	0169F382820	0169A383220	0169B383520	0169C383820	0169D384220	0169E384520	0169F384820	0169A385220	0169B385520		</				



daq file forensic software -- git commit: 04ab90d34f3c86a158d884eca93b366600549046, git branch: dev

spartan configuration files acquisition raw data display plot textual slices summary pmts

pmt bank #12

injection	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	don't inject on any channel
threshold	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	user defined
output	G1	get channels to G1															

serial port configuration

serial port:

iss module info:

module id:

firmware version:

iss mode:

serial number:

serial status: serial port closed

ready to go !!

1 PRÉSENTATION DE L'EXPÉRIENCE

■ objectif et principe de fonctionnement

- Objectifs
- Principe

■ chaîne d'électronique

- schéma bloc
- schéma visuel de l'objet

- la réalisation
- l'avancement

2 ENVIRONNEMENT DE DÉVELOPPEMENT

■ logiciels, bibliothèques, langage

■ besoins et contraintes

■ les logiciels développés

- Les projets et leur dépendances

3 STRATÉGIE DE MUTUALISATION

■ création de bibliothèques

- git
- bibliothèques simples
- bibliothèques hybrides
- bibliothèques externes
- les bibliothèques partagées localement, le cas x2_daq_visu
- la cartographie des relies, le cas tordu ?

4 PRÉSENTATION DE QUELQUES LOGICIELS

■ x2_daq_visu

- spartan
- files
- acquisition
- data display
- plot
- print

5 CONCLUSION

- 
- **pourquoi je vous parle de tout ça ?**
 - un certain nombre des concepts sont transposables
 - firmware avec hdlmake ?
 - le partage pourrait dépasser le cadre de xemis

That's all folks Question?

