

News and Discussion Items

Tom Junk

DUNE Software Management and Architecture Meeting

October 20, 2021

Personnel

- David Adams is moving on from DUNE work.
- We appreciate his immense efforts in DUNE software management, data preparation, ProtoDUNE and Iceberg data monitoring and analysis.
- Jake Calcutt has volunteered to take his role as release manager.

- Etienne Chardonnet, our dunetpc librarian, has defended his thesis and also will be moving on.
- We are looking for a replacement. 2-4 hours per week.
- It's good to have a team of ≥ 2 people, with skills to do the release job, in case one is unavailable for a particular week

HDF5 Status

- Proof-of-concept readin source provided by Kurt, maintained by Tom, used to read Vertical Drift coldbox data into a larsoft job.
- Does not scale to FD, and there is a lot of clumsiness. No PD yet.
- Barnali Chowdhury (ANL) and Jake Calcutt (Oregon State) have volunteered to make a better input source and tool.
 - Division of work between source and tool necessary because of delayed-read functionality.
 - Data product describing the open HDF5 file defined and filled with a new source. Provides communication between source and tool.
 - Channel map work in progress.

dataformats

Source code from the DAQ group:

<https://github.com/DUNE-DAQ/dataformats>

Takes over functionality from `dune_raw_data`

- Dataset header formats
 - WIB frame formats
 - Future: channel maps?
-
- Need this to be available to DAQ, offline builds + running jobs

dataformats

- Currently, the contents are just headers, documentation, and Boost-based tests.
- I made a UPS product out of v3.0.0, not building or testing anything, but copying headers.
- I ran into one difficulty:
 - `#include "ers/Issue.h"`
 - defines a macro for making objects passed to an exception
 - I couldn't locate the ers product – must be a DAQ thing
 - Not hard to replace the exception objects with simpler ones
`throw std::exception "helpful string";`
 - There's also a dependence on daq-cmake but since I wasn't building it, I didn't notice.
- I also named the product "dunedataformats" to avoid name clashes.

dataformats

- Since then, Alessandro Thea has proposed splitting this into three packages
 - **detdataformats** : containing WIB1, WIB2, SSP, DAPHNE .. overlay classes
 - **daqdataformats**: containing TriggerRecordHeader, FragmentHeader, etc.
 - **detchannelmaps**: Hardware to offline channel mappings
- I held off publishing dunedataformats v3_0_0 but the VD Coldbox decoder team (Jake, Barnali, and me) are ready to use the contents.
- Giovanna sent an e-mail today announcing detdataformats and daqdataformats – UPS products to come: dunedetdataformats and dunedaqdataformats
- They also have to go in SciSoft.

Evolution of Data Formats

- Brett brought up the Elephant in the Room: DAQ "has no memory of the past".

While offline must support reading old data, DAQ is mainly concerned with current data.

- Versioning dataformats isn't enough. All old versions must be available to offline jobs.
- An offline job may read more than one file, in different formats, or it may just not know before the file is open what the format of that file is.

Data Format Evolution Solutions

- One solution (as implemented in ICEBERG)
 - DAQ group updates data format
 - DAQ group increments the version number(s) in the data files
 - Offline group forks the DAQ-supplied source code, renames it, builds with all versions in various source files, and provides a switch based on what the version number is to call which deserializer
- Another solution: (mostly implemented in ProtoDUNE-SP)
 - DAQ group provides deserializers with the logic to detect the data format and unpack appropriately
 - ProtoDUNE-SP had RCE, FELIX, compressed/uncompressed, old FELIX metadata/new FELIX metadata, container fragments and non-container fragments. And a lot of personpower that Iceberg lacked.
 - Requirement #3 in DUNE-Doc-4877

A third solution: moo and Kaitai

Suggested by Brett:

- <https://github.com/brettviren/moo>
- <https://kaitai.io/>

ProtoDUNE-SP WIB Frame Format

DUNE-Doc-1701-v3

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ																																
1	Created:	13Sep2016																																																																		
2	Updated:	09/20/2017 (no change for RCE from 08/02/2017)																																																																		
3	Version:	1.1-r2-RCE																																																																		
4		K/D	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Data Source	Notes																																
5		0001	Reserved (8)								SlotNo	CrateNo								FiberNo								Version = 0x1								SOF								WIB	SOF is K2B.5(0xBC) Format version currently = 1.																							
6		0000	WIB Errors																Reserved (14)														OOS	MM	WIB	OOS: bad syncs coming from the PDTs. MM: COLDDATA convert																																
7		0000	Timestamp [62:48] or WIB counter [3]																Timestamp [31:0]														Timestamp [47:32]								WIB	Bits [30:16] of word 3 may be either the upper 8 bits of the tim																										
8		0000	ChkSm B [7:0]								ChkSm A [7:0]								Reserved (8)								Stream 2 ERR								Stream 1 ERR								COLDDATA 1/WIB	WIB-CD1																								
9		0000	COLDDATA Convert Count																																																																	
10		0000	Reserved																																																																	
11		0000	HDR8								HDR6								HDR7								HDR5								HDR4								HDR2								HDR3								HDR1								COLDDATA 1	Header Bits
12		0000	ADC2 CH2[3:0]				ADC2 CH1[11:8]				ADC1 CH2[3:0]				ADC1 CH3[7:0]				ADC1 CH1[11:8]				ADC2 CH1[7:0]				ADC2 CH2[11:4]				ADC1 CH1[7:0]				ADC1 CH2[11:4]				COLDDATA 1																													
13		0000	ADC2 CH3[7:0]				ADC2 CH4[11:4]				ADC1 CH3[7:0]				ADC1 CH4[11:4]				ADC2 CH4[3:0]				ADC2 CH3[11:8]				ADC1 CH4[3:0]				ADC1 CH3[11:8]				COLDDATA 1																																	
14		0000	ADC2 CH6[3:0]				ADC2 CH5[11:8]				ADC1 CH6[3:0]				ADC1 CH5[11:8]				ADC2 CH5[7:0]				ADC2 CH6[11:4]				ADC1 CH5[7:0]				ADC1 CH6[11:4]				COLDDATA 1																																	
15		0000	ADC2 CH7[7:0]				ADC2 CH8[11:4]				ADC1 CH7[7:0]				ADC1 CH8[11:4]				ADC2 CH8[3:0]				ADC2 CH7[11:8]				ADC1 CH8[3:0]				ADC1 CH7[11:8]				COLDDATA 1																																	
16		0000	ADC4 CH2[3:0]				ADC4 CH1[11:8]				ADC3 CH2[3:0]				ADC3 CH1[11:8]				ADC4 CH1[7:0]				ADC4 CH2[11:4]				ADC3 CH1[7:0]				ADC3 CH2[11:4]				COLDDATA 1																																	
17		0000	ADC4 CH3[7:0]				ADC4 CH4[11:4]				ADC3 CH3[7:0]				ADC3 CH4[11:4]				ADC4 CH4[3:0]				ADC4 CH3[11:8]				ADC3 CH4[3:0]				ADC3 CH3[11:8]				COLDDATA 1																																	
18		0000	ADC4 CH6[3:0]				ADC4 CH5[11:8]				ADC3 CH6[3:0]				ADC3 CH5[11:8]				ADC4 CH5[7:0]				ADC4 CH6[11:4]				ADC3 CH5[7:0]				ADC3 CH6[11:4]				COLDDATA 1																																	
19		0000	ADC4 CH7[7:0]				ADC4 CH8[11:4]				ADC3 CH7[7:0]				ADC3 CH8[11:4]				ADC4 CH8[3:0]				ADC4 CH7[11:8]				ADC3 CH8[3:0]				ADC3 CH7[11:8]				COLDDATA 1																																	
20		0000	ADC6 CH2[3:0]				ADC6 CH1[11:8]				ADC5 CH2[3:0]				ADC5 CH1[11:8]				ADC6 CH1[7:0]				ADC6 CH2[11:4]				ADC5 CH1[7:0]				ADC5 CH2[11:4]				COLDDATA 1																																	
21		0000	ADC6 CH3[7:0]				ADC6 CH4[11:4]				ADC5 CH3[7:0]				ADC5 CH4[11:4]				ADC6 CH4[3:0]				ADC6 CH3[11:8]				ADC5 CH4[3:0]				ADC5 CH3[11:8]				COLDDATA 1																																	
22		0000	ADC6 CH6[3:0]				ADC6 CH5[11:8]				ADC5 CH6[3:0]				ADC5 CH5[11:8]				ADC6 CH5[7:0]				ADC6 CH6[11:4]				ADC5 CH5[7:0]				ADC5 CH6[11:4]				COLDDATA 1																																	
23		0000	ADC6 CH7[7:0]				ADC6 CH8[11:4]				ADC5 CH7[7:0]				ADC5 CH8[11:4]				ADC6 CH8[3:0]				ADC6 CH7[11:8]				ADC5 CH8[3:0]				ADC5 CH7[11:8]				COLDDATA 1																																	
24		0000	ADC8 CH2[3:0]				ADC8 CH1[11:8]				ADC7 CH2[3:0]				ADC7 CH1[11:8]				ADC8 CH1[7:0]				ADC8 CH2[11:4]				ADC7 CH1[7:0]				ADC7 CH2[11:4]				COLDDATA 1																																	
25		0000	ADC8 CH3[7:0]				ADC8 CH4[11:4]				ADC7 CH3[7:0]				ADC7 CH4[11:4]				ADC8 CH4[3:0]				ADC8 CH3[11:8]				ADC7 CH4[3:0]				ADC7 CH3[11:8]				COLDDATA 1																																	
26		0000	ADC8 CH6[3:0]				ADC8 CH5[11:8]				ADC7 CH6[3:0]				ADC7 CH5[11:8]				ADC8 CH5[7:0]				ADC8 CH6[11:4]				ADC7 CH5[7:0]				ADC7 CH6[11:4]				COLDDATA 1																																	
27		0000	ADC8 CH7[7:0]				ADC8 CH8[11:4]				ADC7 CH7[7:0]				ADC7 CH8[11:4]				ADC8 CH8[3:0]				ADC8 CH7[11:8]				ADC7 CH8[3:0]				ADC7 CH7[11:8]				COLDDATA 1																																	
28		0000	ChkSm B [7:0]								ChkSm A [7:0]								Reserved (8)								Stream 2 ERR								Stream 1 ERR								COLDDATA 2/WIB	WIB-CD2																								
29		0000	COLDDATA Convert Count																																																																	
30		0000	Reserved																																																																	
31		0000	HDR8								HDR6								HDR7								HDR5								HDR4								HDR2								HDR3								HDR1								COLDDATA 2	Header Bits
32		0000	ADC2 CH2[3:0]				ADC2 CH1[11:8]				ADC1 CH2[3:0]				ADC1 CH3[7:0]				ADC1 CH1[11:8]				ADC2 CH1[7:0]				ADC2 CH2[11:4]				ADC1 CH1[7:0]				ADC1 CH2[11:4]				COLDDATA 2																													
33		0000	ADC2 CH3[7:0]				ADC2 CH4[11:4]				ADC1 CH3[7:0]				ADC1 CH4[11:4]				ADC2 CH4[3:0]				ADC2 CH3[11:8]				ADC1 CH4[3:0]				ADC1 CH3[11:8]				COLDDATA 2																																	
34		0000	ADC2 CH6[3:0]				ADC2 CH5[11:8]				ADC1 CH6[3:0]				ADC1 CH5[11:8]				ADC2 CH5[7:0]				ADC2 CH6[11:4]				ADC1 CH5[7:0]				ADC1 CH6[11:4]				COLDDATA 2																																	
35		0000	ADC2 CH7[7:0]				ADC2 CH8[11:4]				ADC1 CH7[7:0]				ADC1 CH8[11:4]				ADC2 CH8[3:0]				ADC2 CH7[11:8]				ADC1 CH8[3:0]				ADC1 CH7[11:8]				COLDDATA 2																																	
36		0000	ADC2 CH2[3:0]				ADC2 CH1[11:8]				ADC1 CH2[3:0]				ADC1 CH3[7:0]				ADC1 CH1[11:8]				ADC2 CH1[7:0]				ADC2 CH2[11:4]				ADC1 CH1[7:0]				ADC1 CH2[11:4]				COLDDATA 2																													
37		0000	ADC2 CH3[7:0]				ADC2 CH4[11:4]				ADC1 CH3[7:0]				ADC1 CH4[11:4]				ADC2 CH4[3:0]				ADC2 CH3[11:8]				ADC1 CH4[3:0]				ADC1 CH3[11:8]				COLDDATA 2																																	
38		0000	ADC2 CH6[3:0]				ADC2 CH5[11:8]				ADC1 CH6[3:0]				ADC1 CH5[11:8]				ADC2 CH5[7:0]				ADC2 CH6[11:4]				ADC1 CH5[7:0]				ADC1 CH6[11:4]				COLDDATA 2																																	
39		0000	ADC2 CH7[7:0]				ADC2 CH8[11:4]				ADC1 CH7[7:0]				ADC1 CH8[11:4]				ADC2 CH8[3:0]				ADC2 CH7[11:8]				ADC1 CH8[3:0]				ADC1 CH7[11:8]				COLDDATA 2																																	
40		0000	ChkSm B [7:0]								ChkSm A [7:0]								Reserved (8)								Stream 2 ERR								Stream 1 ERR								COLDDATA 2	16 bit frame checksum will be calculated for each 8 bit wide str																								
41		0000	COLDDATA Convert Count																																																																	
42		0000	Reserved																																																																	
43		0000	HDR8								HDR6								HDR7								HDR5								HDR4								HDR2								HDR3								HDR1								COLDDATA 2	Header Bits
44		0000	ADC2 CH2[3:0]				ADC2 CH1[11:8]				ADC1 CH2[3:0]				ADC1 CH3[7:0]				ADC1 CH1[11:8]				ADC2 CH1[7:0]				ADC2 CH2[11:4]				ADC1 CH1[7:0]				ADC1 CH2[11:4]				COLDDATA 2																													
45		0000	ADC2 CH3[7:0]				ADC2 CH4[11:4]				ADC1 CH3[7:0]				ADC1 CH4[11:4]				ADC2 CH4[3:0]				ADC2 CH3[11:8]				ADC1 CH4[3:0]				ADC1 CH3[11:8]				COLDDATA 2																																	
46		0000	ADC2 CH6[3:0]				ADC2 CH5[11:8]				ADC1 CH6[3:0]				ADC1 CH5[11:8]				ADC2 CH5[7:0]				ADC2 CH6[11:4]				ADC1 CH5[7:0]				ADC1 CH6[11:4]				COLDDATA 2																																	
47		0000	ADC2 CH7[7:0]				ADC2 CH8[11:4]				ADC1 CH7[7:0]				ADC1 CH8[11:4]				ADC2 CH8[3:0]				ADC2 CH7[11:8]				ADC1 CH8[3:0]				ADC1 CH7[11:8]				COLDDATA 2																																	

Just the first few rows. 256 channels (FELIX link)

Crate, slot, fiber numbers set by the WIB. Used to identify channels.

HDF5 dataset headers have redundant geometrical information. What to do if they conflict?

Channel map status

- Nitish gave me a pointer to a spreadsheet with strip ID's and cold electronics information for the Bottom Drift Electronics
- Slavic committed a VD Coldbox gdml and a channel sorter
- I made wire dumps here:
https://cdcv.sfnal.gov/redmine/projects/dunetpc/wiki/_dunevdc_b1_Wire_Dumps_
- I have a text file that associates strips with offline channels.
- Need to write an *art* service that reads this file and provides the map. Where to put it? `dune-raw-data`? `dunetpc`? `detchannelmaps`? It needs an ecosystem to read that file. Or to even be a service.
- Some cold electronics channels are not connected to any strip. Not sure where to put the data in the geometry.

Changing getByLabel to getHandle

- Kyle Knoepfel announced in April that `art::Event::getByLabel` is obsolete and will be deprecated soon
- Deprecation is worse – it creates a compiler warning which we treat as an error, halting the build.
- Over 530 instances of `getByLabel` in `dunetpc` and `protoduneana`. Another batch ~100 in `garsoft`.
- I converted the ones in `garsoft` and `protoduneana`, and started on `dunetpc`
- Still 300 instances left in `dunetpc`.
- I've been modifying `develop` directly but should use a feature branch.
- Not scriptable. Changes declarations, and sometimes control flow (in simple ways).
- `getManyByType` got removed in `art v3.0.9`

getByLabel → getHandle

- Do not use getByLabel in the future – code new getHandles
- Instructions are here:

<https://indico.fnal.gov/event/48717>

Splitting dunetpc

- Scripts are ready; they had been waiting for Dom to get back from holiday and release the newly refactored FD fcl files.
- I got sidetracked by other projects and haven't pushed on that.
- To do:
 - Untangle fcls in unit tests so each product at least can be tested with only dependent products set up. DUNE software is supposed to work with the whole stack, so not every fcl knot needs to be untied at this stage.
 - update scripts to preserve git history. At the moment they just cp source to new repositories. That trims the size, but makes people go to redmine to get history.
 - update Jenkins build script
 - coordinate with Vito about testing all the new repos
 - Request new repo directories in SciSoft
 - Disable pushes to dunetpc in Redmine
 - deploy in GitHub
 - Request mrb to pull repos out of GitHub and not Redmine

Meeting Time

- Is Wednesday at 1:00 PM Central Time good? It's not too great for Europeans
- Our previous slot was Weds at 2:00 PM
- Mornings Central Time are pretty busy on DUNE