INTT offline software

sPHENIX calibration

Intt Dead channel map generator

- Hot channel is masked out, i.e. it's treated as dead channel
- https://github.com/sPHENIX-Collaboration/calibrations/blob/master/Tracking/INTT/macros/Construct_DeadMap.C

Intt Dead channel map:

- One for each layer in XML format. Totally 4 layers
- https://github.com/sPHENIX-Collaboration/calibrations/tree/master/Tracking/INTT/DeadMap.

Copy to sPHENIX???

sPHENIX coresoftware

Intt Geant4 simulation

https://github.com/sPHENIX-Collaboration/coresoftware/tree/master/simulation/g4simulation/g4intt

Intt offline

https://github.com/sPHENIX-Collaboration/coresoftware/tree/master/offline/packages/intt

CylinderGeomIntt.cc/.h

- Define INTT geometry, e.g. strip spacing, strip center local coordinate.

InttClusterizer.cc/.h

- create INTT clusters. Directly call or transplant to online monitoring

Intt Geant4 simulation

https://github.com/sPHENIX-Collaboration/coresoftware/tree/master/simulation/g4simulation/g4intt

PHG4InttDefs.h
- Indices to identify intt components

PHG4InttFPHXParameterisation.cc/.h
- FPHX geometry parameter

PHG4InttDetector.cc/.h
- Build intt g4 detector

PHG4InttDisplayAction.cc/.h
- Intt display settings

PHG4InttSteppingAction.cc/.h
- Trace each step??

PHG4InttSubsystem.cc/.h
- Run intt g4 sim

InttDeadMap.cc/.h & InttDeadMapv1.cc/.h & PHG4InttDeadMapLoader.cc/.h
- Load dead channel map.

PHG4InttHitReco.cc/.h
- Find fired channels

PHG4InttDigitizer.cc/.h
- Hit adc

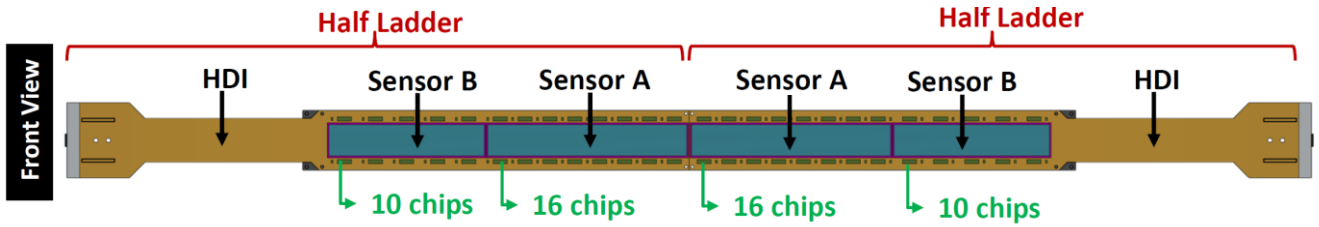
The software need to be cleaned up (later). For example,
- Still has the Z-sensitive layer
- In PHG4InttHitReco.cc:

  "// I made this (small) diffusion up for now, we will get actual values for the Intt later
  double diffusion_width = 5.0e-04; // diffusion radius 5 microns, in cm "

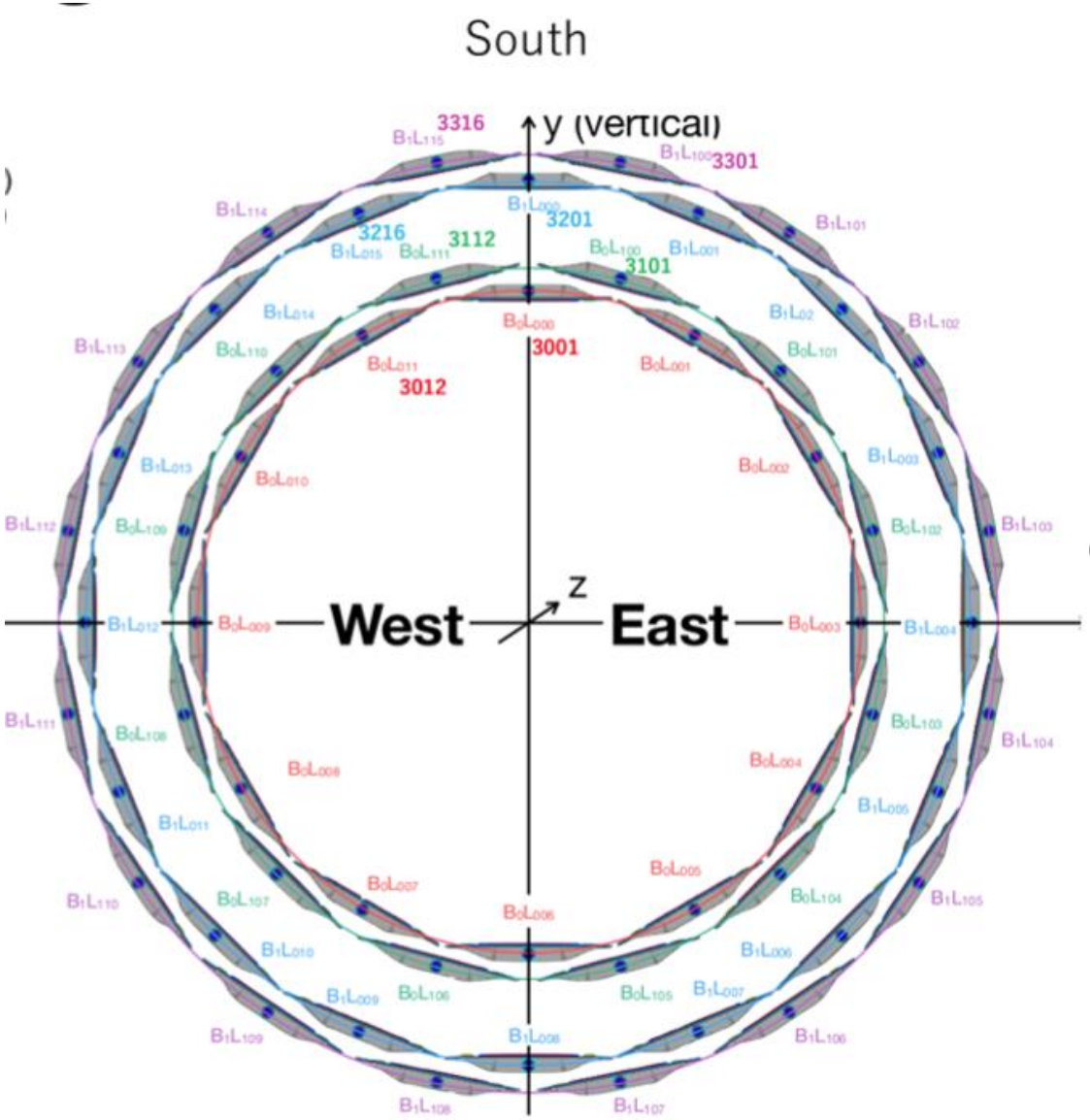
INTT channel location: offline convention

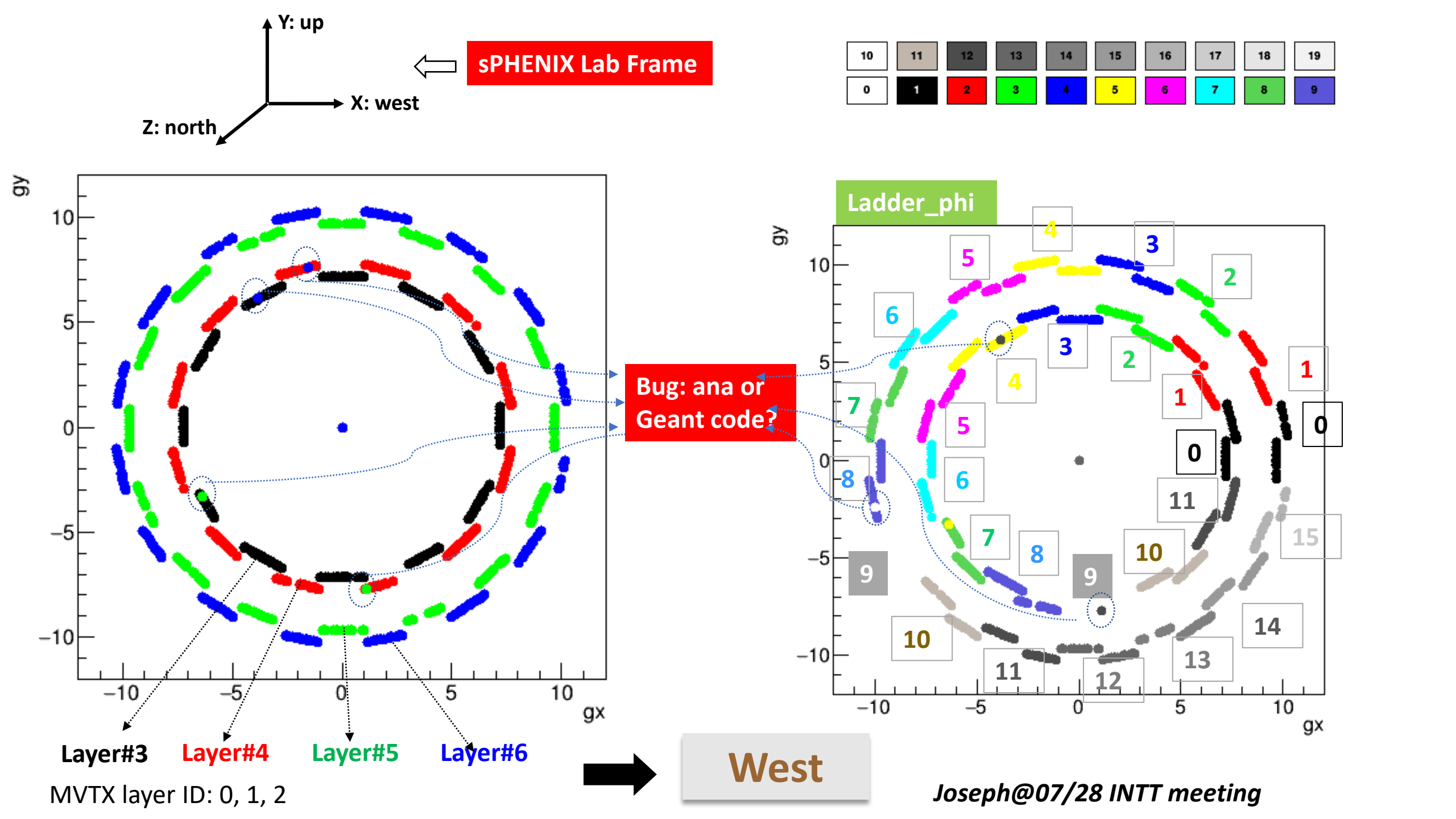
sPHENIX offline convention to determine a single strip is through the following indices:

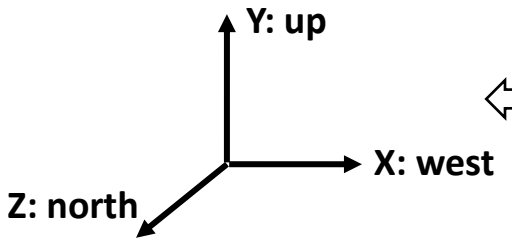
- **Layer**
  - layer index 0-3, i.e. 4 layers, or 3-4 when MVTX is included
- **ladder_phi**
  - ladder index in each layer (0-11)@B0L0, (0-11)@B0L1, (0-15)@B1L0, (0-15)@B1L1
- **ladder_z**
  - sensor index in each ladder 0-4, i.e. 4 sensors



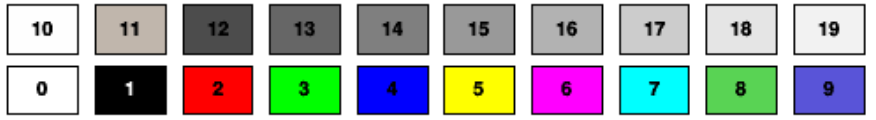
- **strip_z**
  - strip index along Z in each sensor, i.e. (0-7), (0-4), (0-7), (0-4)
- **strip_phi**
  - strip index along phi in each sensor, i.e. (0-255)

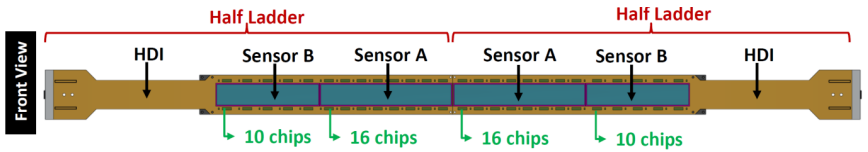



Y: up
X: west
Z: north
sPHENIX Lab Frame
10 11 12 13 14 15 16 17 18 19
0 1 2 3 4 5 6 7 8 9
gy
gx
Ladder_phi
Bug: ana or Geant code?
Layer#3
Layer#4
Layer#5
Layer#6
MVTX layer ID: 0, 1, 2
West
Joseph@07/28 INTT meeting



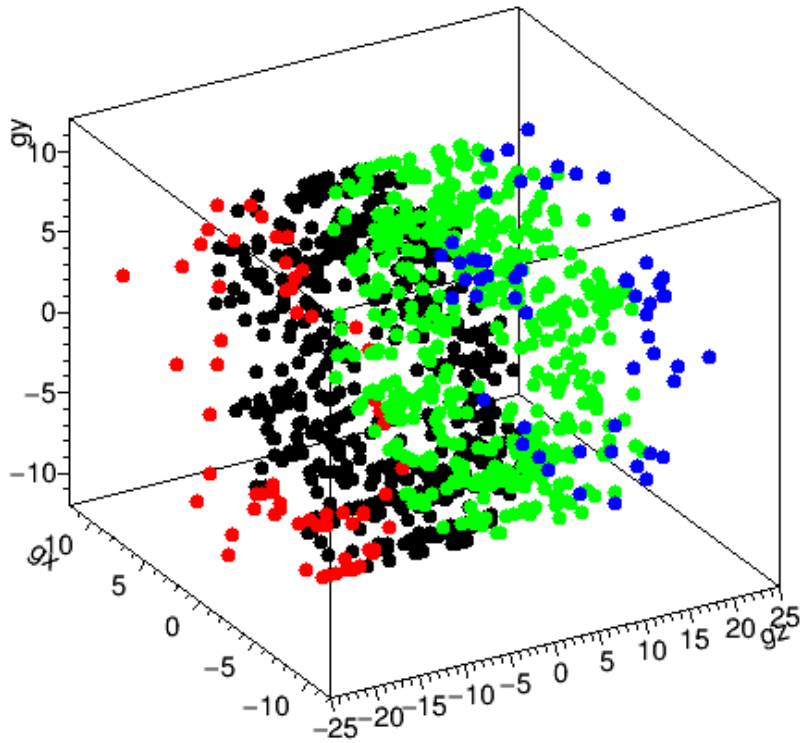
sPHENIX Lab Frame



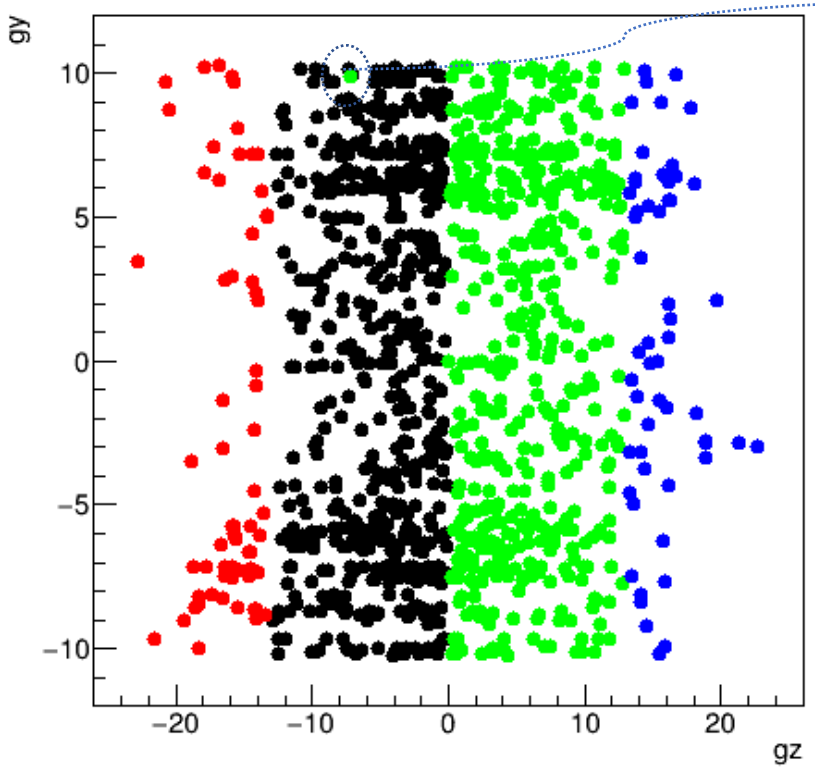


1 0 2 3

Ladder_z

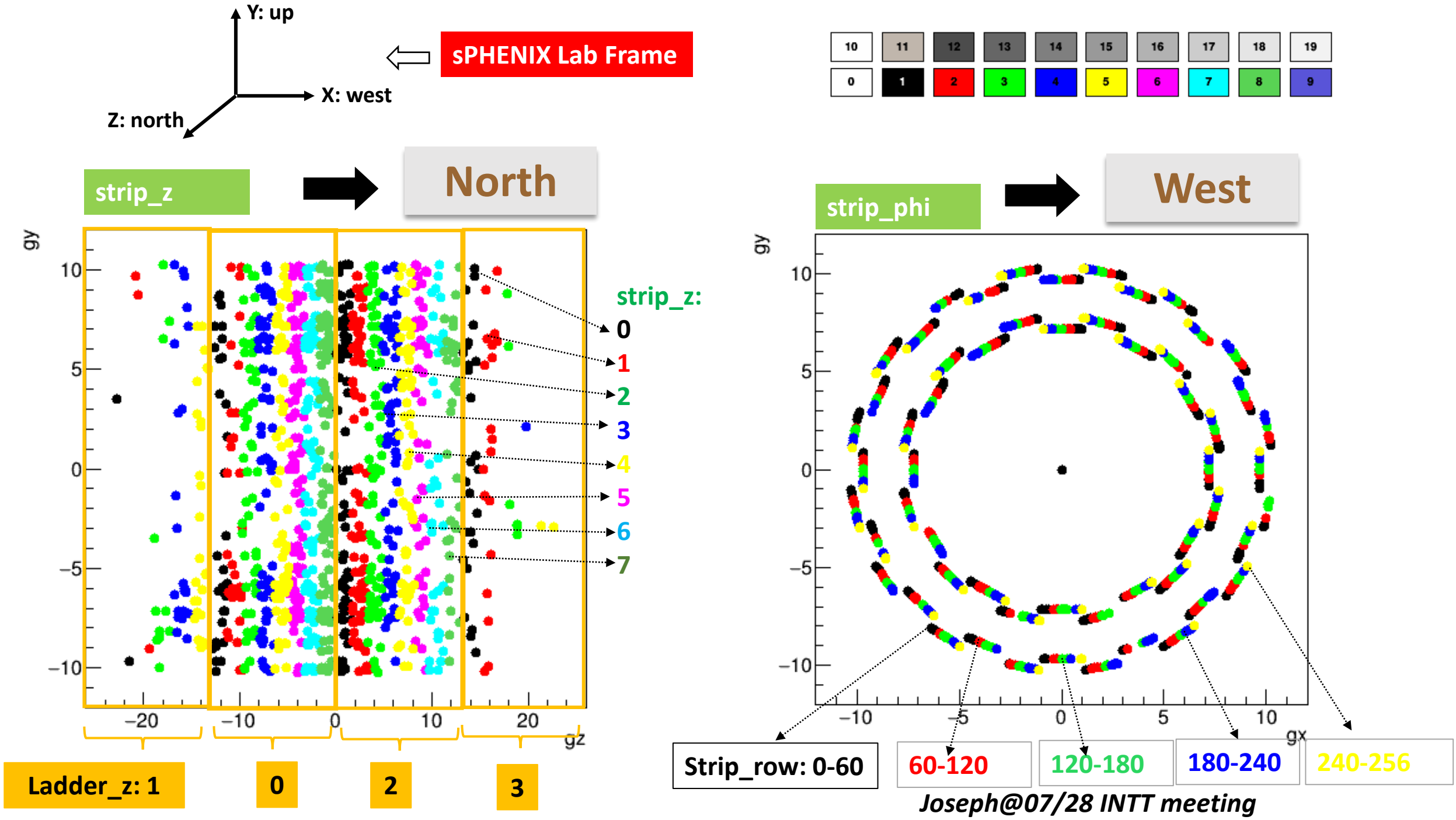


Bug: ana or Geant code?



North

Joseph@07/28 INTT meeting


Y: up
X: west
Z: north
sPHENIX Lab Frame
10 11 12 13 14 15 16 17 18 19
0 1 2 3 4 5 6 7 8 9
strip_z
North
strip_phi
West
gy
gz
gx
strip_z:
0
1
2
3
4
5
6
7
Ladder_z: 1
0
2
3
Strip_row: 0-60
60-120
120-180
180-240
240-256
Joseph@07/28 INTT meeting

Martin will write the unpacker and we will help to test it

From: Martin Purschke <purschke@bnl.gov>
Date: Jul 21, 2022 10:55 PM
Subject: Re: FVTX raw data format
To: "Xie, Wei" <wxie@purdue.edu>
Cc:

---- External Email: Use caution with attachments, links, or sharing data ----

Hi Wei,

it doesn't quite work that way...

It is true that at the lowest "core" of what a ROC delivers is a 64bit word with several pieces of info encoded, like BCO, CHIPID, ADC, ROW, etc etc. However, the FELIX then packages this info in a way that is quite different from the PHENIX FVTX (that was read out and packaged in that case by the DCM2, completely different device and firmware). There will be per-ROC headers and other headers added by the FELIX, and in general it is not known in detail yet how the data will arrive.

Raul is still working on and finding the best organization in the FELIX firmware, so when I say "not known yet" I mean this literally.

I wrote the unpacker for the FVTX back then, and I don't see a way around me doing the same again for the INTT - that is not something I am comfortable delegating all the way (I would very much like to enlist some help for the *testing* when the time comes). To begin with, Raul and I can sit together side by side and quickly clarify whatever is not clear, but very often I need to ask for changes in the format because, for example, some crucial information is missing or ambiguous. BTW, the INTT is in no way special here - I have written all the unpackers (except one) in the history of PHENIX and sPHENIX.

I discussed this a bit with Joseph already; he indicated that he would be ready to help and test as much as possible. I'm assuming you asked him contact me and offer that help.

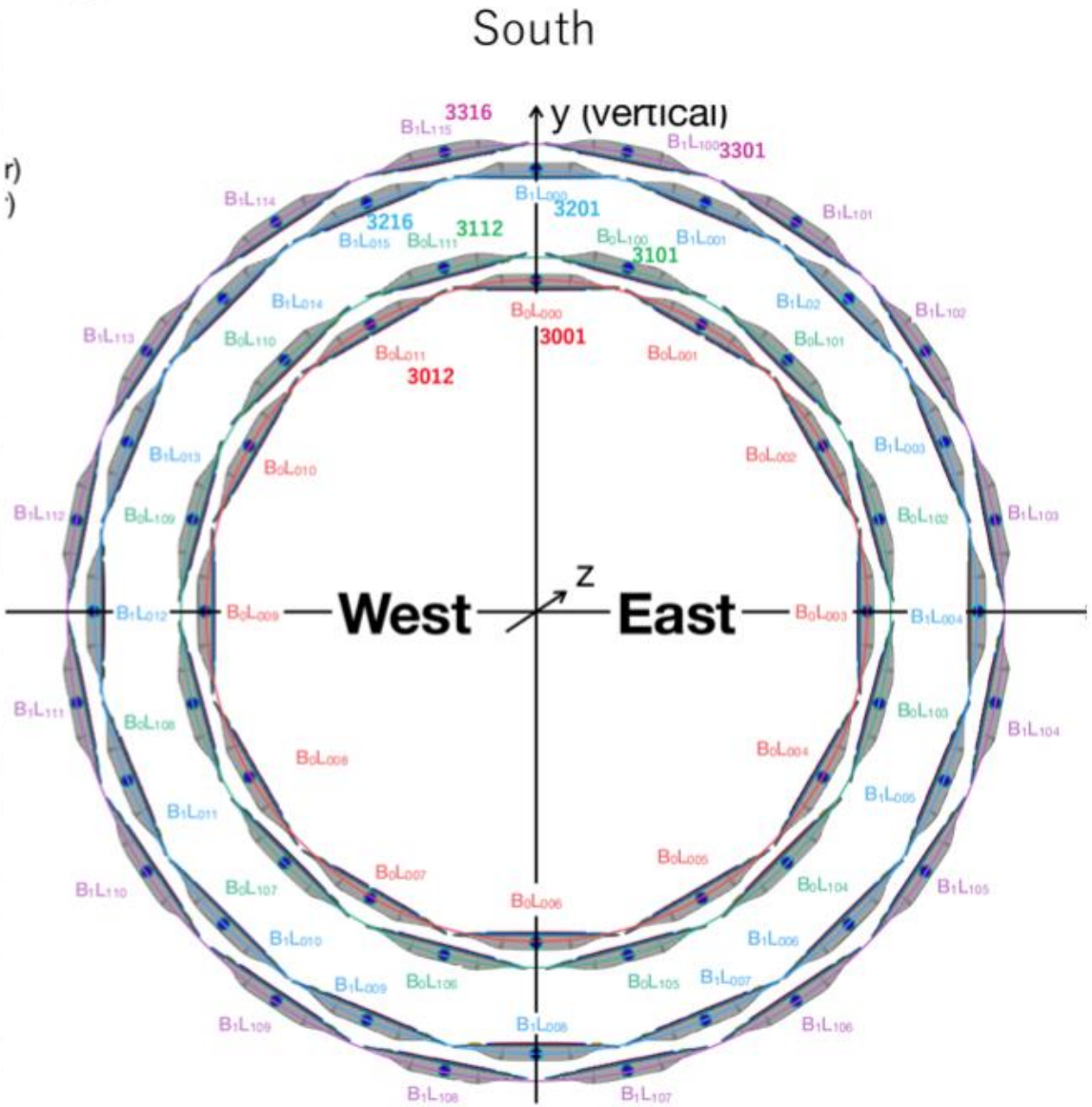
So, patience - I have a RCDAQ plugin going that gets whatever the FELIX is giving me, just as I said the contents of that are still very much in flux.

Hope it helps...

Martin

Proposed Packet-ID Assignment

Ladder	South	North	Ladder	South	North
B0L000	3001	3501	B0L100	3101	3601
B0L001	3002	3502	B0L101	3102	3602
B0L002	3003	3503	B0L102	3103	3603
B0L003	3004	3504	B0L103	3104	3604
B0L004	3005	3505	B0L104	3105	3605
B0L005	3006	3506	B0L105	3106	3606
B0L006	3007	3507	B0L106	3107	3607
B0L007	3008	3508	B0L107	3108	3608
B0L008	3009	3509	B0L108	3109	3609
B0L009	3010	3510	B0L109	3110	3610
B0L010	3011	3511	B0L110	3111	3611
B0L011	3012	3512	B0L111	3112	3612
Ladder	South	North	Ladder	South	North
B1L000	3201	3701	B1L100	3301	3801
B1L001	3202	3702	B1L101	3302	3802
B1L002	3203	3703	B1L102	3303	3803
B1L003	3204	3704	B1L103	3304	3804
B1L004	3205	3705	B1L104	3305	3805
B1L005	3206	3706	B1L105	3306	3806
B1L006	3207	3707	B1L106	3307	3807
B1L007	3208	3708	B1L107	3308	3808
B1L008	3209	3709	B1L108	3309	3809
B1L009	3210	3710	B1L109	3310	3810
B1L010	3211	3711	B1L110	3311	3811
B1L011	3212	3712	B1L111	3312	3812
B1L012	3213	3713	B1L112	3313	3813
B1L013	3214	3714	B1L113	3314	3814
B1L014	3215	3715	B1L114		
BLL015	3216	3716	BLL115		



Proposed **chip ID** and **row** assignment