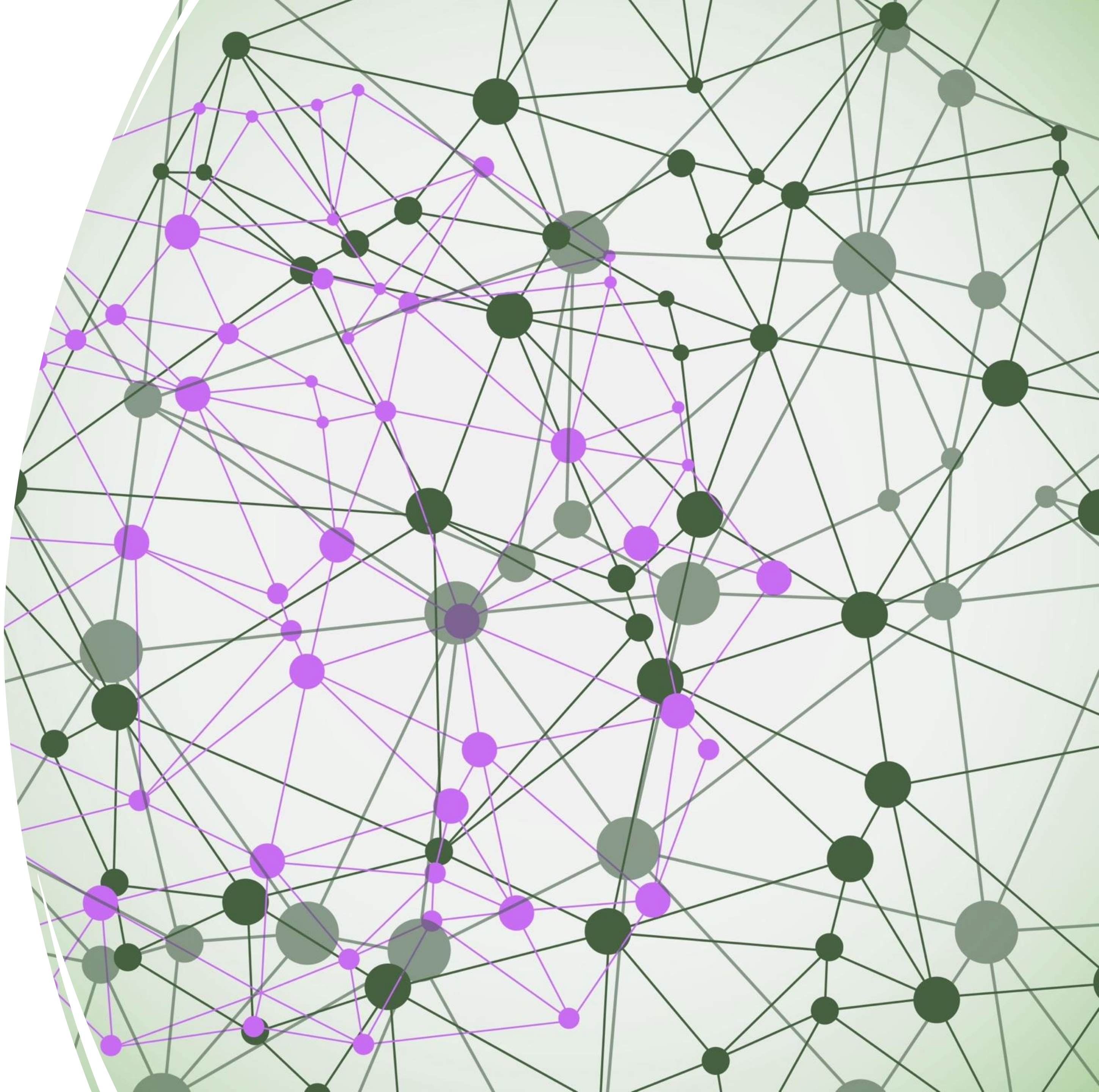
Intelligent Experiments through Real-Time AI:

Fast Data Processing and Autonomous Detector Control for High-Energy Nuclear Experiments

Ming Liu for the Fast-ML Team

Los Alamos National Lab

11/19/2025



The Team – NP, HEP, CS and EE

❑ **A joint effort of NP, HEP, CS and EE**

- **LANL, MIT, FNAL, NJIT, GIT, ORNL et al**

❑ **Physics simulation and AI/ML algorithms**

❑ **Firmware implementation**

- ***hls4ml*, FlowGNN etc.**

❑ **Demonstrator deployment**

- ***FPGA, GPU, CPU etc.***

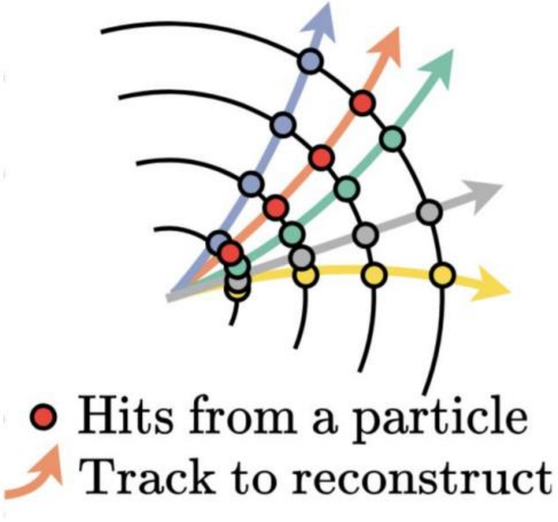
✓ 2022 – 2023, $1.4M

✓ 2024 – 2025, $1.6M

Why Fast-ML?

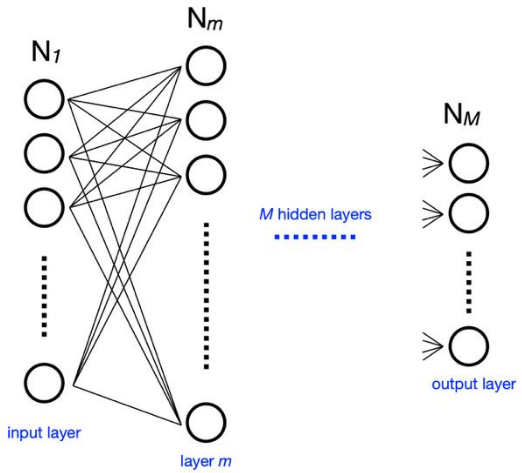
❑ High data throughput from modern detectors in high-energy nuclear experiments

- ➢ O(1)TB/s @detectors, CMS, ATLAS, ALICE, **sPHENIX, EIC** ...
- ➢ Very large data volume (~100PB/year), it also takes a long time to process the data offline for physics analysis



❑ **Our goal** - use AI/ML based algorithms to tag important (rare) events in real-time with high efficiency in p+p and e+p/A collisions, for fast data filtering/reduction

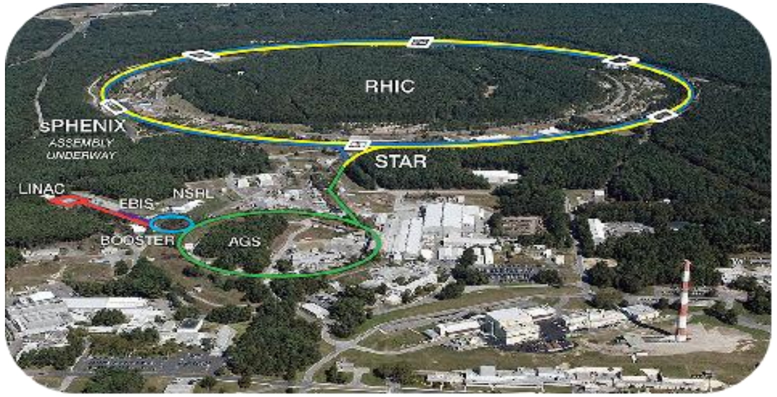
sPHENIX as the first test ground, ultimately for EIC in 2030s

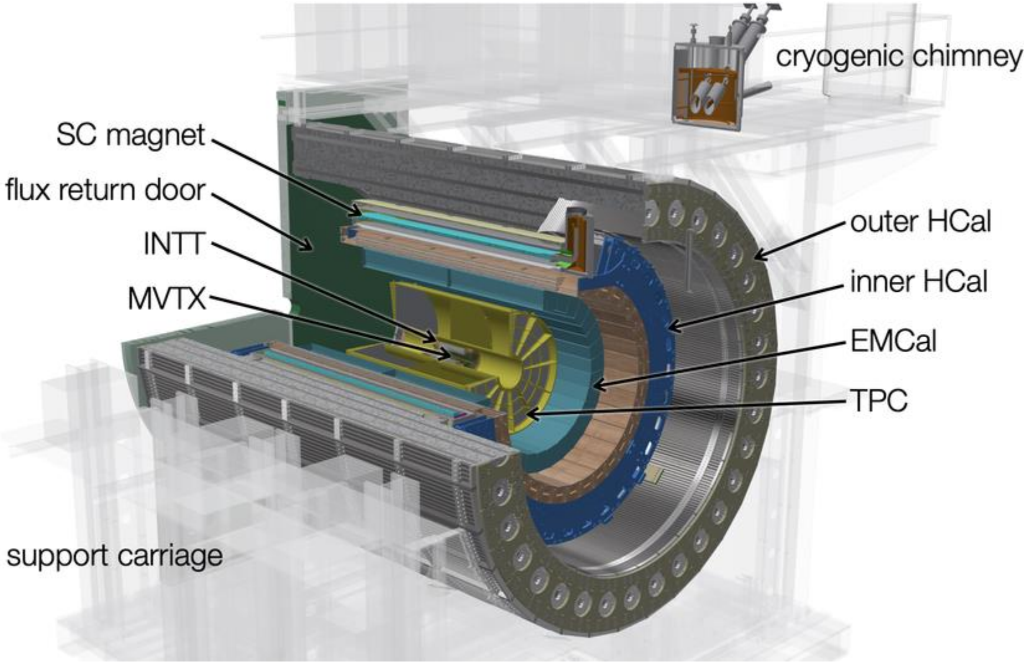


Real-time AI

sPHENIX Experiment at Relativistic Heavy Ion Collider

- Located at RHIC (BNL)
- Running period 2023-2025+
- Main detectors: **tracking detectors** (MVTX, INTT, TPC), calorimeters (EMCal, HCal)
- Hybrid trigger scheme
  - Tracking detectors support streaming readout
    - DAQ limit < 300Gb/sec
  - Calorimeters readout is trigger-based: 15kHz event rate

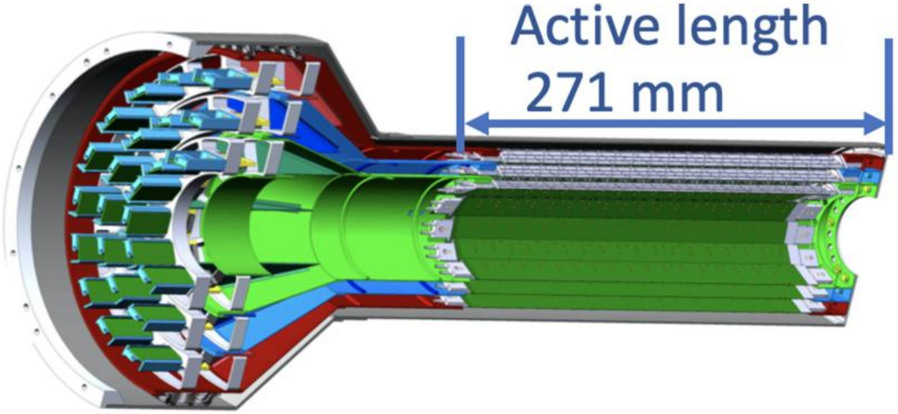




MVTX and INTT: first full streaming readout @RHIC

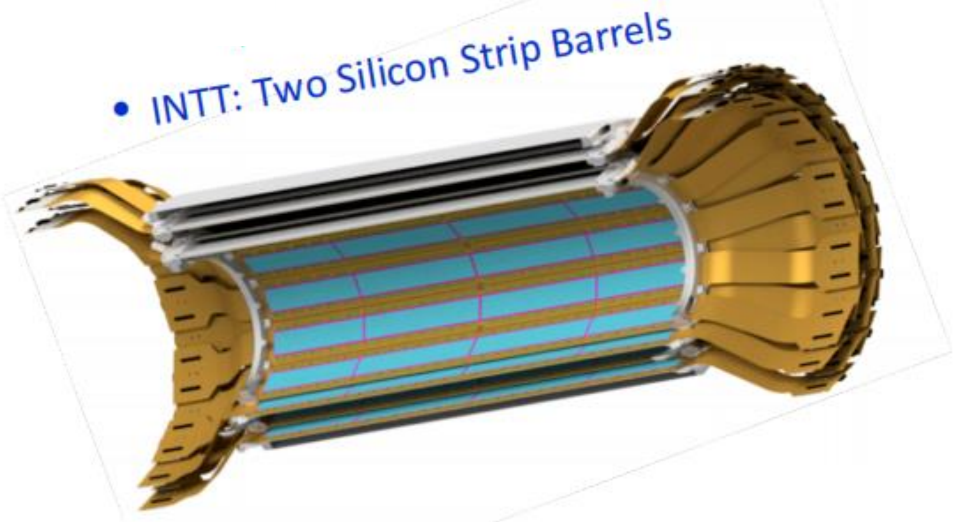
❑ MVTX – Monolithic-active-pixel-sensor based vertex detector

- ➢ Pitch: 27 μm × 29 μm
- ➢ Time resolution: 5 μs
- ➢ 3 layers, 48 staves: ~230M pixels channels



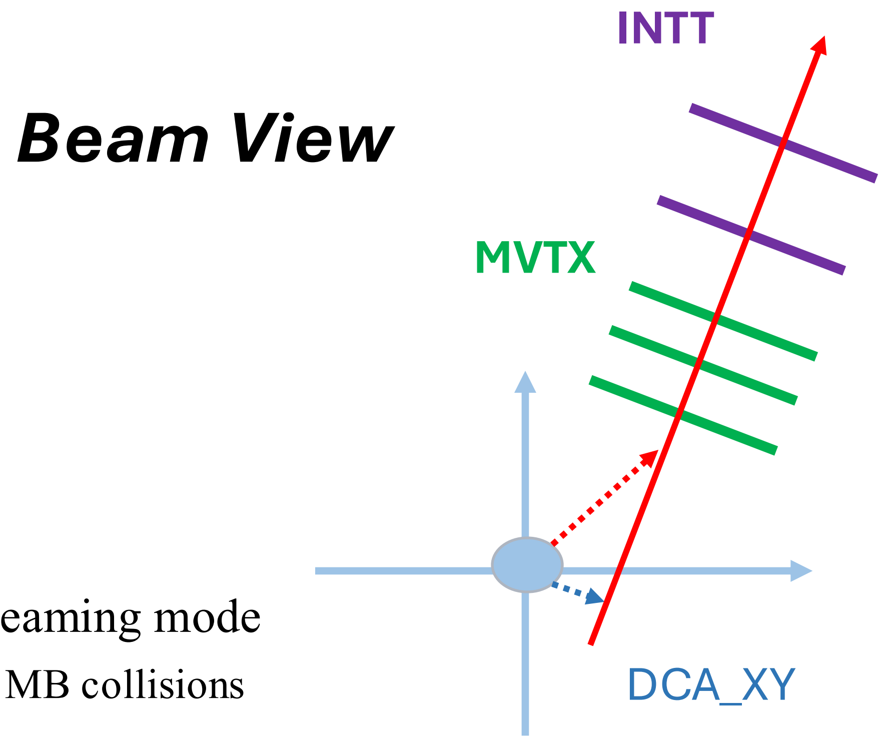
❑ INTT – micro-strip tracking detector

- ➢ Pitch: 27 μm × 16 (or 20) mm
- ➢ Time resolution: ~ 50 ns (< BCO 106ns)
- ➢ 2 layers, 56 ladders



A Test Case: Tag Rare Heavy Quark Events in Real-Time

- *High p+p* collision rate ~2MHz, a lot of data!
  - Charm quark production: ~ 30 kHz
    - 500 μb/42mb ~1%
  - Beauty quarks: ~ 150 Hz
    - 2 μb/42mb ~0.005%
  - sPHENIX DAQ trigger rate: ~<15 kHz
    - Tracking detectors are Streamed Readout (SRO) capable
    - Limited DAQ bandwidth prohibits taking all TPC raw data in full streaming mode
      - TPC working in "trigger + extended" readout mode(~20us), ~O(10%) of MB collisions
    - MVTX and INTT, full SRO in p+p run



- A real-time AI/ML trigger system aiming to tag rare HF events with minimal impacts on overall data throughput, with high purity and efficiency

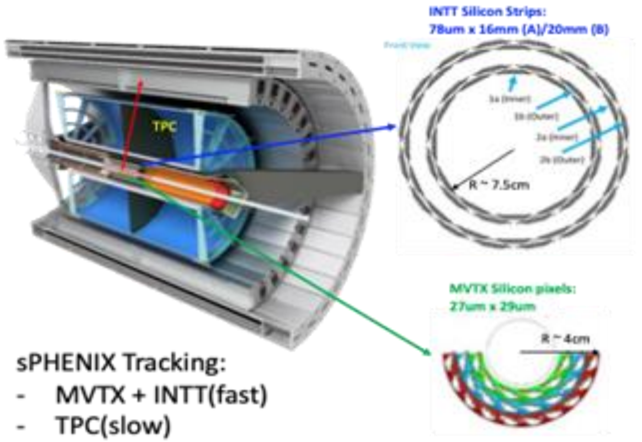
  Note: MB trigger highly pre-scaled, <0.5% total events (~10kHz/2MHz)

Our Playground

- Heavy flavor event AI-trigger demonstrator in sPHENIX

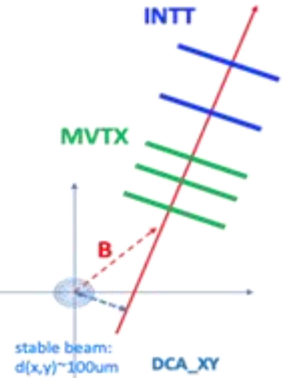
Two half-barrels for trigger decisions

Selective streaming real-time AI and autonomous detector control:
Deliver a demonstrator for p+p and p+A running for sPHENIX - generalizable for applications in experiments at the EIC



sPHENIX Tracking:
- MVTX + INTT(fast)
- TPC(slow)

Identify B-hadron event:
- Topology of B decay, with large DCA
- Monitor collision point



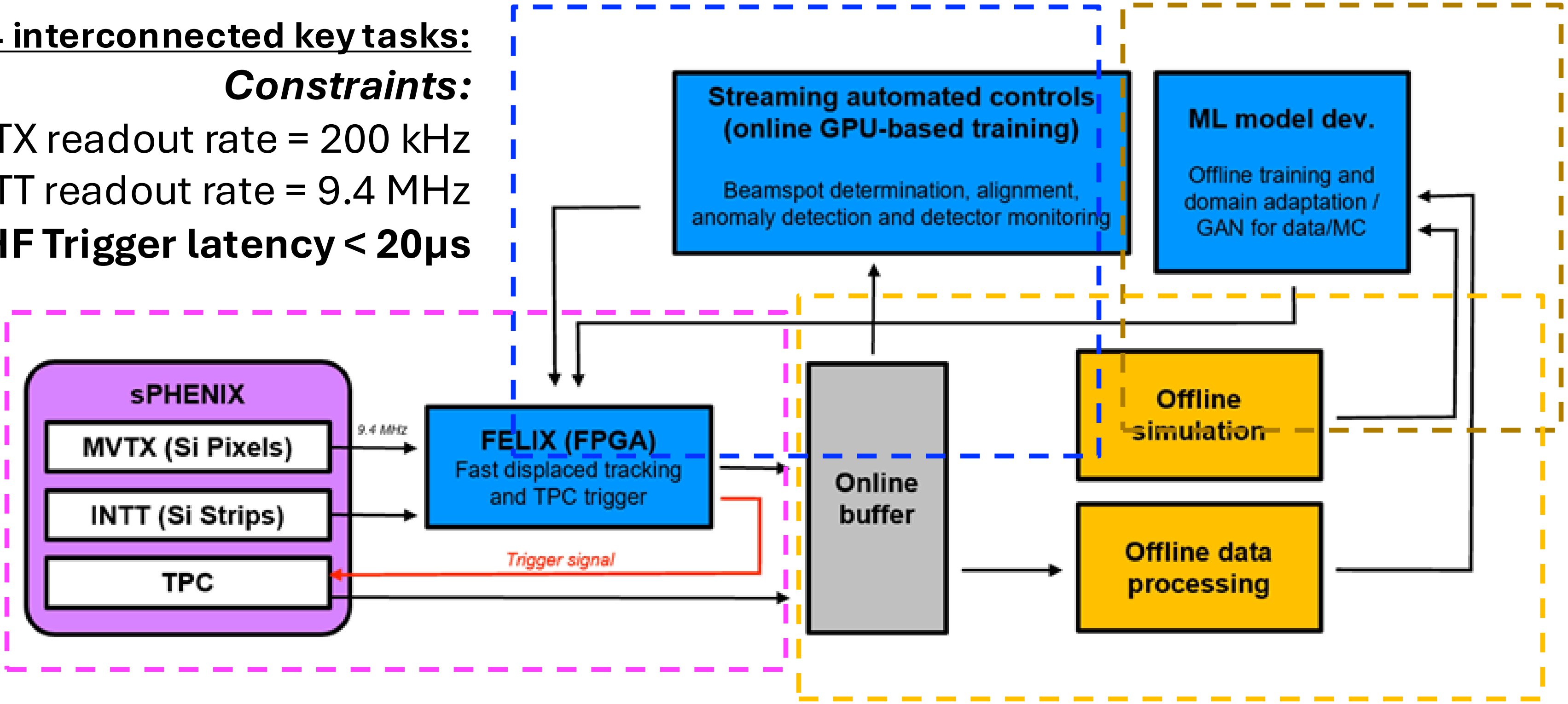
<u>**4 interconnected key tasks:**</u>

Constraints:

MVTX readout rate = 200 kHz

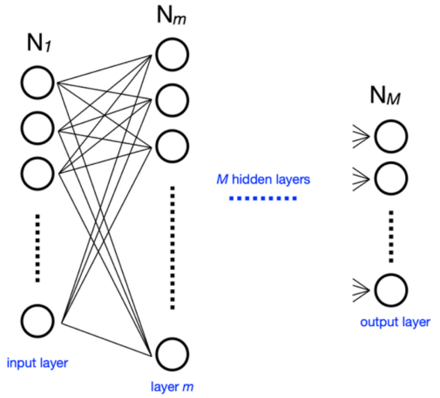
INTT readout rate = 9.4 MHz

HF Trigger latency < 20μs


Streaming automated controls (online GPU-based training)
Beamspot determination, alignment, anomaly detection and detector monitoring
ML model dev.
Offline training and domain adaptation / GAN for data/MC
sPHENIX
MVTX (Si Pixels)
INTT (Si Strips)
TPC
9.4 MHz
FELIX (FPGA)
Fast displaced tracking and TPC trigger
Trigger signal
Online buffer
Offline simulation
Offline data processing

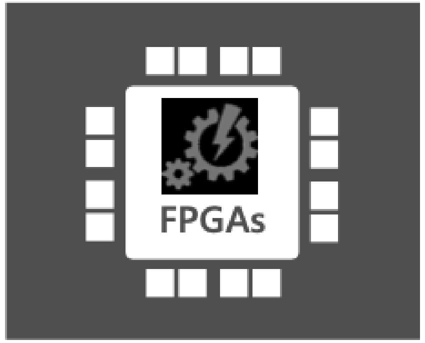
3 Successful R&D Focus Areas

- demonstrated the feasibility



talk by Dantong

- ❑ Physics/detector simulations and AI/ML algorithm development



talk by Callie

- ❑ Translate AI/ML algorithms into hardware language FPGA code with (1) data processing latency and (2)hardware resource constraints

- ❑ Deploy FPGA algorithms in a demonstrator system in sPHENIX

talk by Jovan

Lessons learned from sPHENIX operation with real beams, p+p, Au+Au in 2023-2025

talk by Cameron/Yasser

Potential implications for the EIC (early 2030s)

Physics simulation and AI-ML algorithms

- Dantong Yu and Giorgian Borca-Tasciuc (NJIT)

Approach

❑ Machine learning model family trained on two separate objectives

  ➢ D^0 and Beauty Decay Events in p+p collisions

❑ Bipartite architecture design for accurate and low-latency inference

  ➢ Graph-Convolutional Neural Network (GCN) used as baseline

❑ Two Separate approaches are used:

  ➢ Single-Stage: Hits➔Trigger Detection

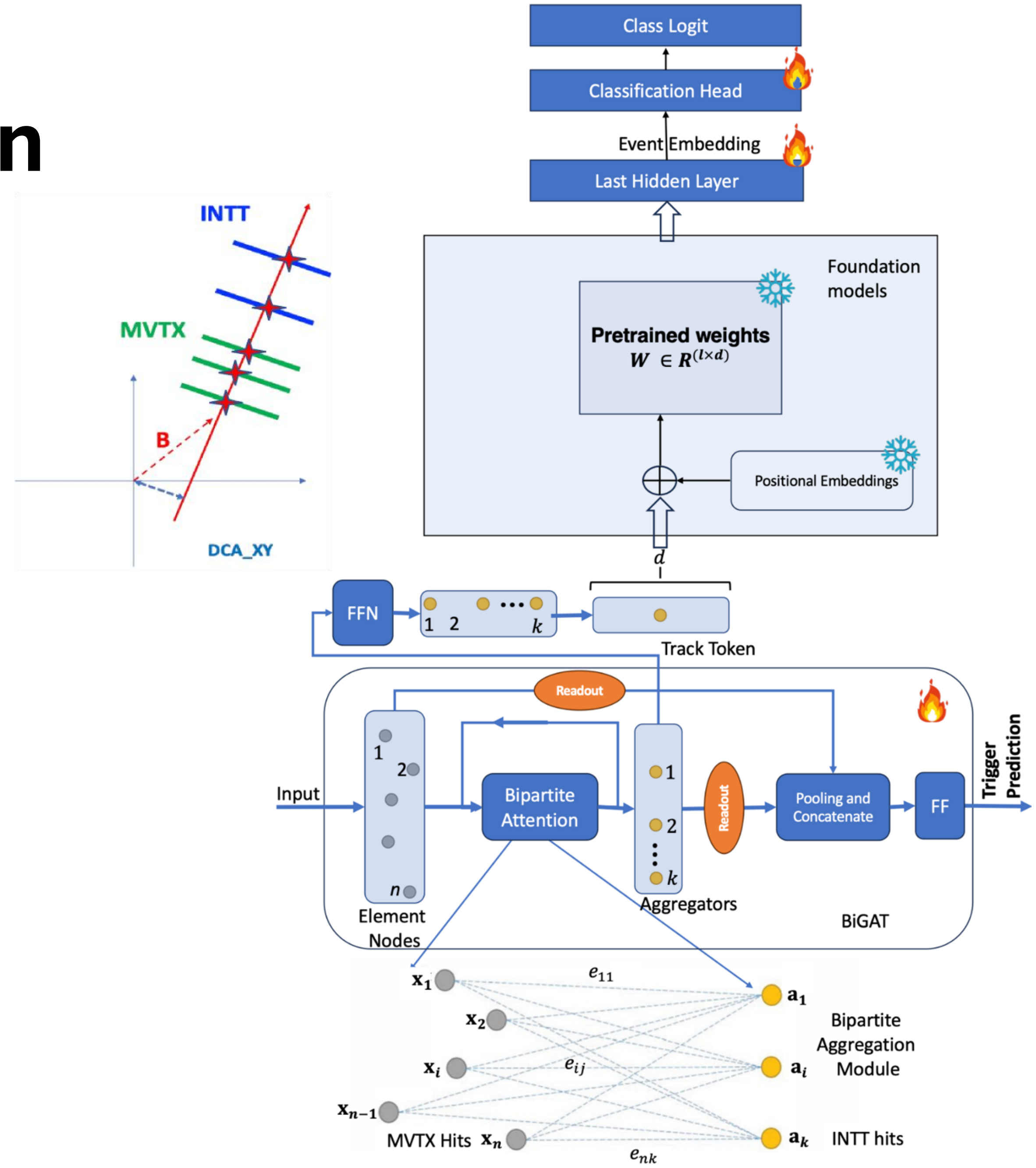
  ➢ Two-Stage: Hits➔Tracking➔Trigger Detection

❑ Performance of tracking stage and trigger stage quantified separately

❑ Effect of event pileup is included and evaluated

Bipartite Architecture Design

- ❑Align with Detector Hardware
  - ➢MVTX v.s. INTT (50ns time stamps)
  - ➢Use INTT as anchors to reject out out-of-sync MVTX hits
- ❑Tracking between INTT and MVTX
- ❑Bipartite graph represents all feasible connections between MVTX hits on the left and INTT hits on the right.
- ❑Frozen Foundational Attention Networks



Tracking Performance

F1 score = harmonic mean of precision/recall

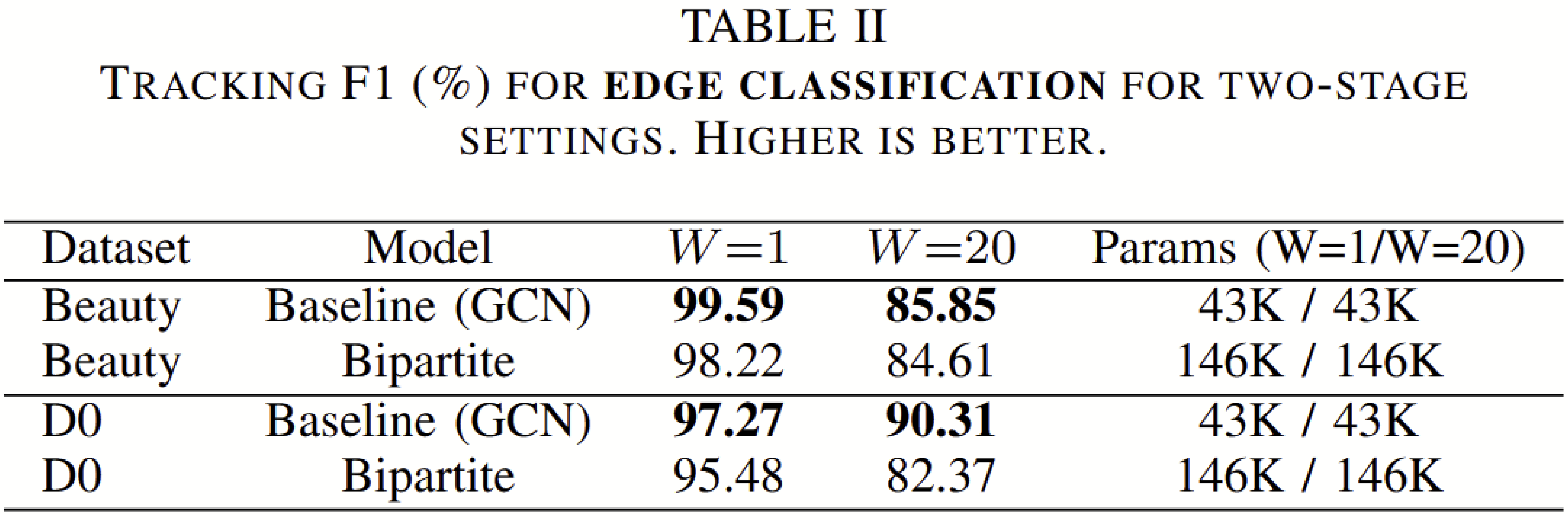


TABLE II
TRACKING F1 (%) FOR **EDGE CLASSIFICATION** FOR TWO-STAGE SETTINGS. HIGHER IS BETTER.

Dataset	Model	$W=1$	$W=20$	Params (W=1/W=20)
Beauty	Baseline (GCN)	**99.59**	**85.85**	43K / 43K
Beauty	Bipartite	98.22	84.61	146K / 146K
D0	Baseline (GCN)	**97.27**	**90.31**	43K / 43K
D0	Bipartite	95.48	82.37	146K / 146K

- GCN model retains advantage in tracking problem
- Bipartite model offers small reduction in tracking performance for large improvement in latency (later slide #24)

AI/ML HF Trigger Performance

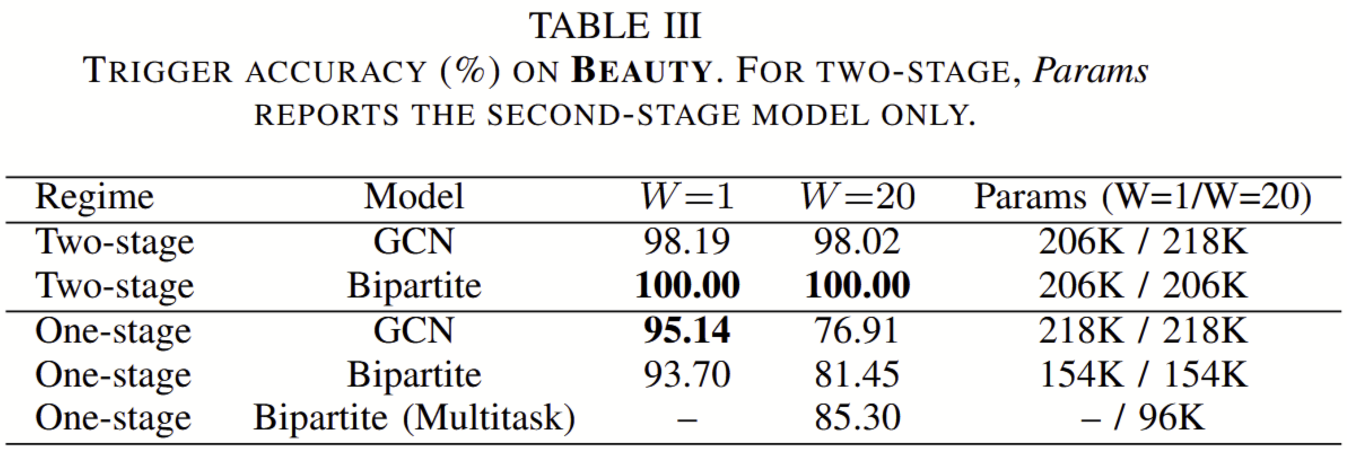


TABLE III

TRIGGER ACCURACY (%) ON **BEAUTY**. FOR TWO-STAGE, *Params* REPORTS THE SECOND-STAGE MODEL ONLY.

Regime	Model	*W*=1	*W*=20	Params (W=1/W=20)
Two-stage	GCN	98.19	98.02	206K / 218K
Two-stage	Bipartite	**100.00**	**100.00**	206K / 206K
One-stage	GCN	**95.14**	76.91	218K / 218K
One-stage	Bipartite	93.70	81.45	154K / 154K
One-stage	Bipartite (Multitask)	–	85.30	– / 96K

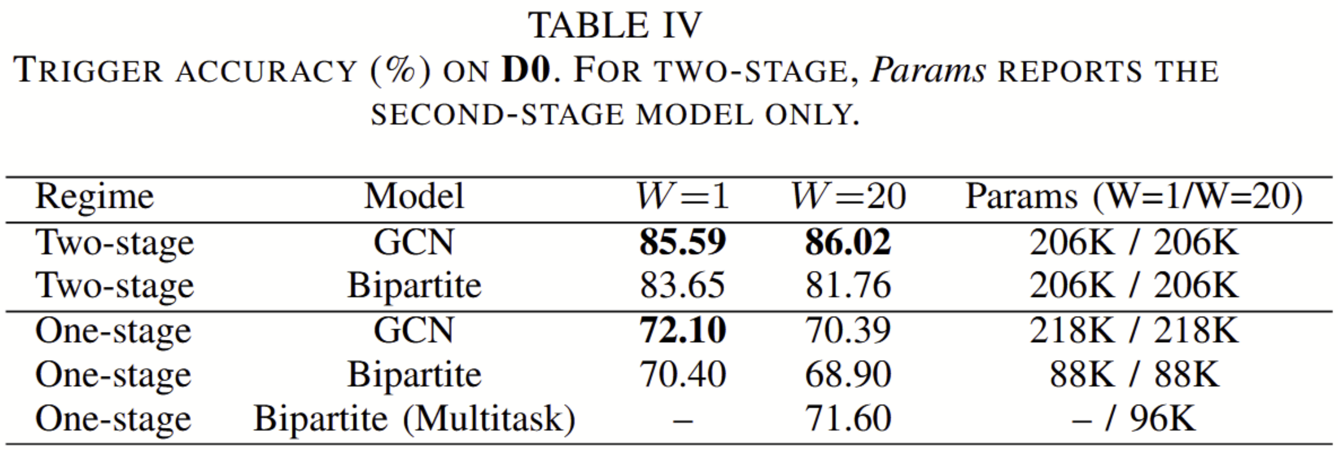


TABLE IV

TRIGGER ACCURACY (%) ON **D0**. FOR TWO-STAGE, *Params* REPORTS THE SECOND-STAGE MODEL ONLY.

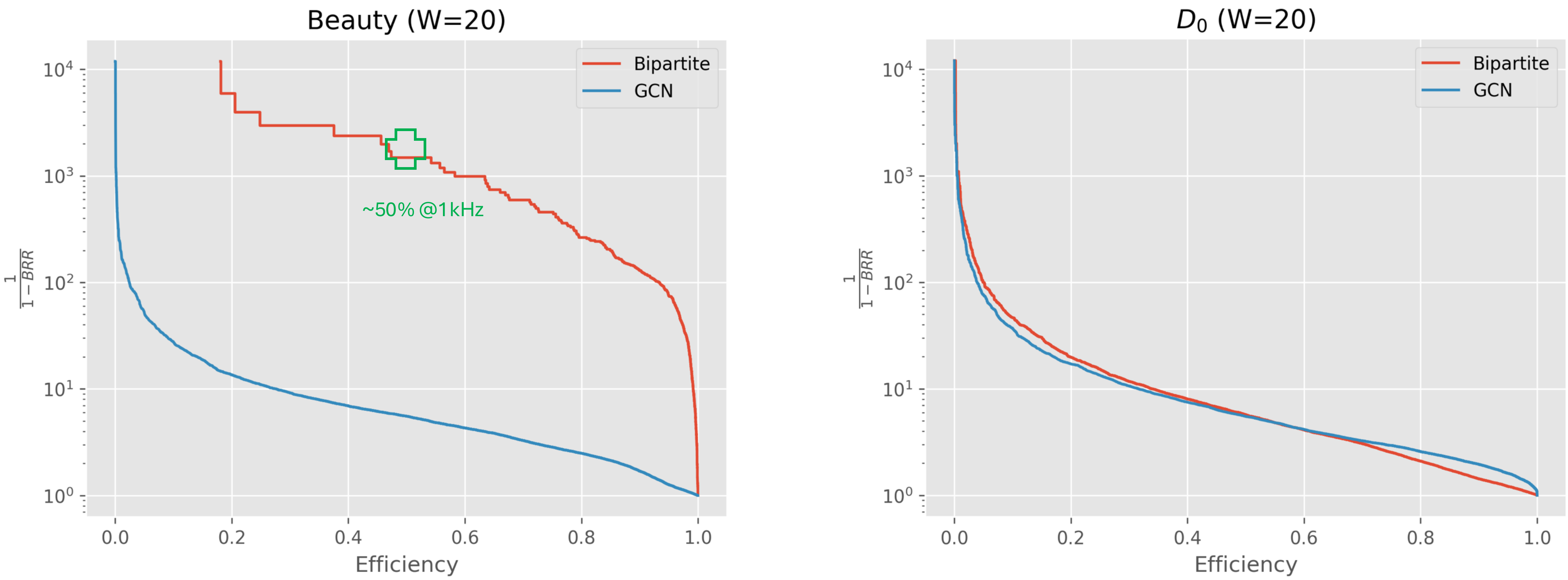
Regime	Model	*W*=1	*W*=20	Params (W=1/W=20)
Two-stage	GCN	**85.59**	**86.02**	206K / 206K
Two-stage	Bipartite	83.65	81.76	206K / 206K
One-stage	GCN	**72.10**	70.39	218K / 218K
One-stage	Bipartite	70.40	68.90	88K / 88K
One-stage	Bipartite (Multitask)	–	71.60	– / 96K

❑Excellent accuracy achieved on two-stage classification of beauty decays with bipartite.

❑Single-Stage bipartite retains >85% accuracy on beauty with pileup.

❑Small total model size (<100K param high-performing models trained)

Signal Efficiency vs Background Reject Rate (BRR)



Latency Evaluation

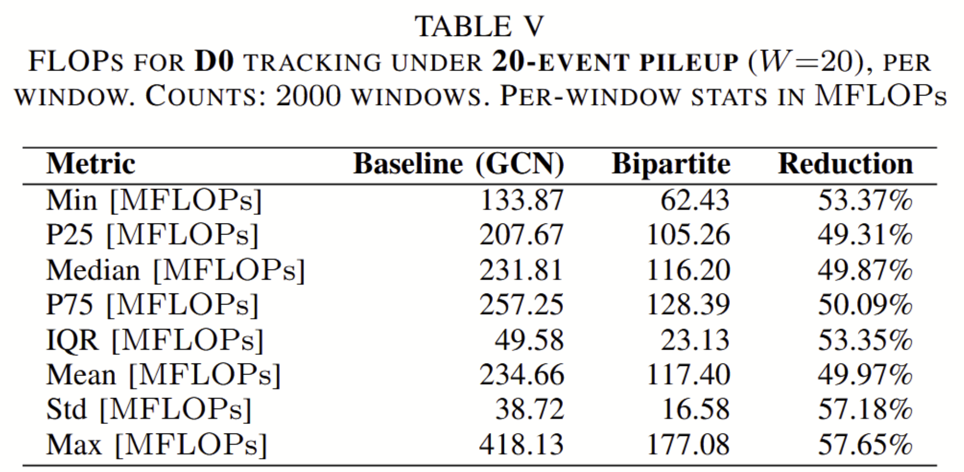


TABLE V
FLOPs for D0 tracking under 20-event pileup (W=20), per window. Counts: 2000 windows. Per-window stats in MFLOPs

Metric	Baseline (GCN)	Bipartite	Reduction
Min [MFLOPs]	133.87	62.43	53.37%
P25 [MFLOPs]	207.67	105.26	49.31%
Median [MFLOPs]	231.81	116.20	49.87%
P75 [MFLOPs]	257.25	128.39	50.09%
IQR [MFLOPs]	49.58	23.13	53.35%
Mean [MFLOPs]	234.66	117.40	49.97%
Std [MFLOPs]	38.72	16.58	57.18%
Max [MFLOPs]	418.13	177.08	57.65%

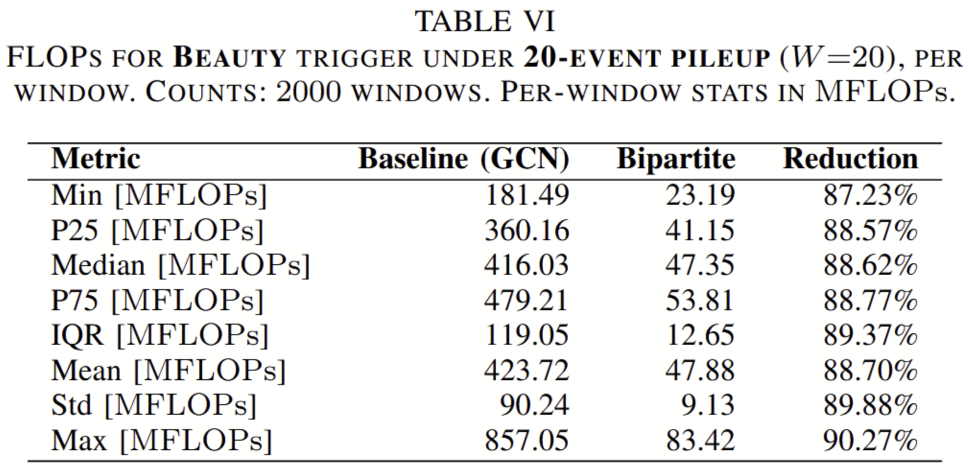
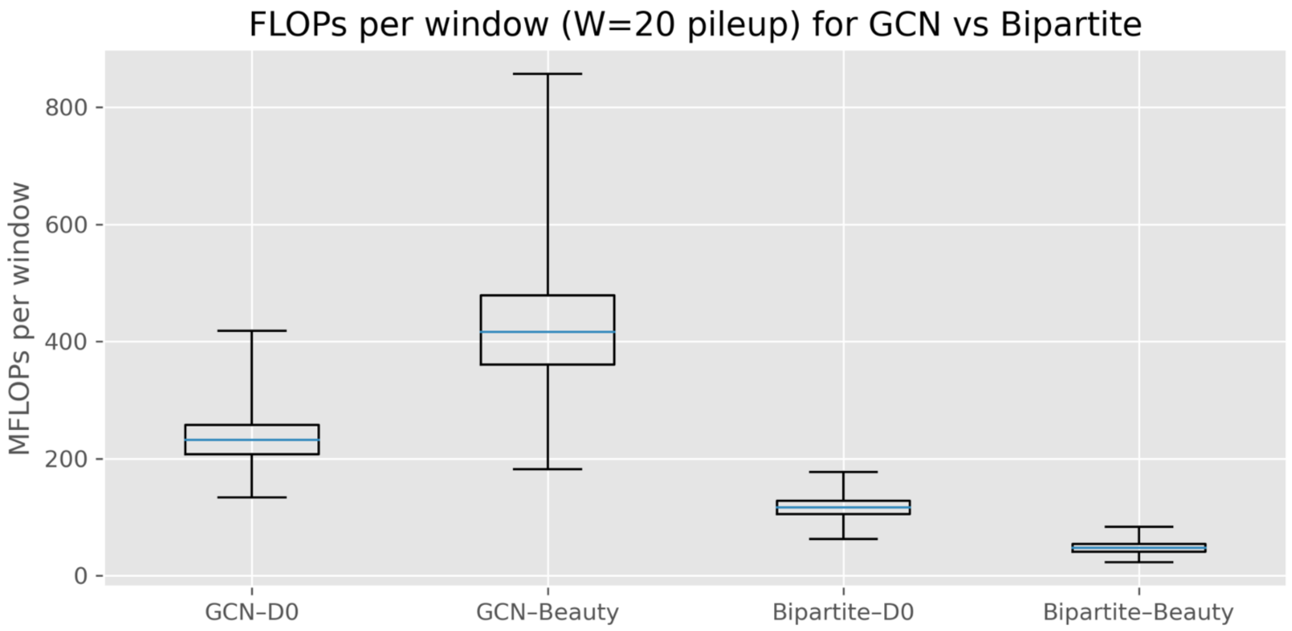


TABLE VI
FLOPs for Beauty trigger under 20-event pileup (W=20), per window. Counts: 2000 windows. Per-window stats in MFLOPs.

Metric	Baseline (GCN)	Bipartite	Reduction
Min [MFLOPs]	181.49	23.19	87.23%
P25 [MFLOPs]	360.16	41.15	88.57%
Median [MFLOPs]	416.03	47.35	88.62%
P75 [MFLOPs]	479.21	53.81	88.77%
IQR [MFLOPs]	119.05	12.65	89.37%
Mean [MFLOPs]	423.72	47.88	88.70%
Std [MFLOPs]	90.24	9.13	89.88%
Max [MFLOPs]	857.05	83.42	90.27%



- ❑ Bipartite model achieves **>50%** reduction in FLOPs in D^0 setting and **>85%** reduction in FLOPs for Beauty setting
  - ➢ This is despite GCN model having **3x fewer params**
  - ➢ **Larger** performance improvement expected for models with **equal** parameter count
- ❑ Bipartite model also achieves a 57%-89% reduction in the spread of latencies and a 90% reduction in the max FLOPs
  - ➢ Important for controlling time-outs

Recap : AI/ML HF Trigger Algorithms

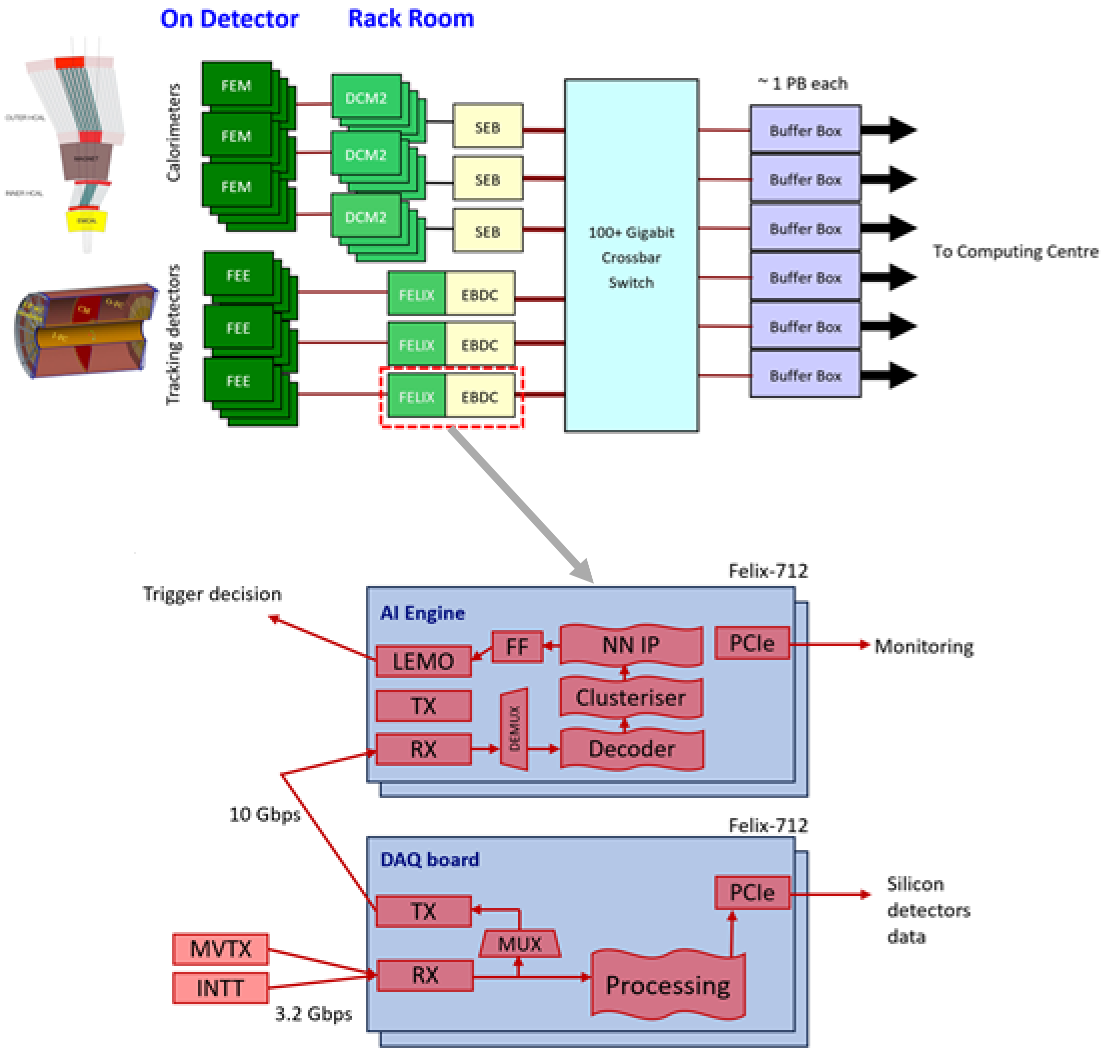
- ❑Bipartite architecture shows **demonstrated order-of magnitude improvements** in FLOP count while achieving **competitive performance**
    - ➢Achieves 8% point improvement in single-stage trigger accuracy for beauty decay event
- ❑Pipeline approach proves **tunable trade-off** between latency and accuracy
- ❑Bipartite model reduces both **maximum** FLOPs and **spread** in FLOPs, allowing more predictable performance
- ❑Approach generalizes across **multiple physics channels**, as demonstrated by competitive performance in both beauty and D^0 classification problems

hls4ml translation and firmware implementation

- Calli Hao (GIT)
- Hannah Bossi (MIT)

Readout and HF AI-Trigger Implementation in FPGA

- The sPHENIX tracking detectors use FELIX-712 PCIe-based boards
  - Contain an AMD/Xilinx Kintex UltraScale FPGA (xcku115-flvf1924-2-e)
- To the readout DAQ boards, add AI Engine boards to perform the B-tagging using AI (FELIX-712)
- Exploring implement graph neural networks (GNNs) with two approaches:
  - FlowGNN (arXiv: 2204.13103)
  - hls4ml (arXiv: 1804.06913)
  - BiPartite model



The Latency Constrains for ML-base Algorithm

- The TPC buffer can hold up ~30 us of data before receiving a readout trigger
- Detector readout delay, fiber transmission delay, data encoding/decoding
  - MVTX readout window ~8us
  - Interaction Region (IR) ->Counting house ~0.3 us (100 m cables)
  - FELIX data forward, decoder buffers ~0.6 us (@240 MHz)
  - Global level 1 Trigger decision latency + counting house -> IR ~0.3 us
  - Total **~10 us**
- The goal is to achieve **~10 us** latency for the trigger algorithm

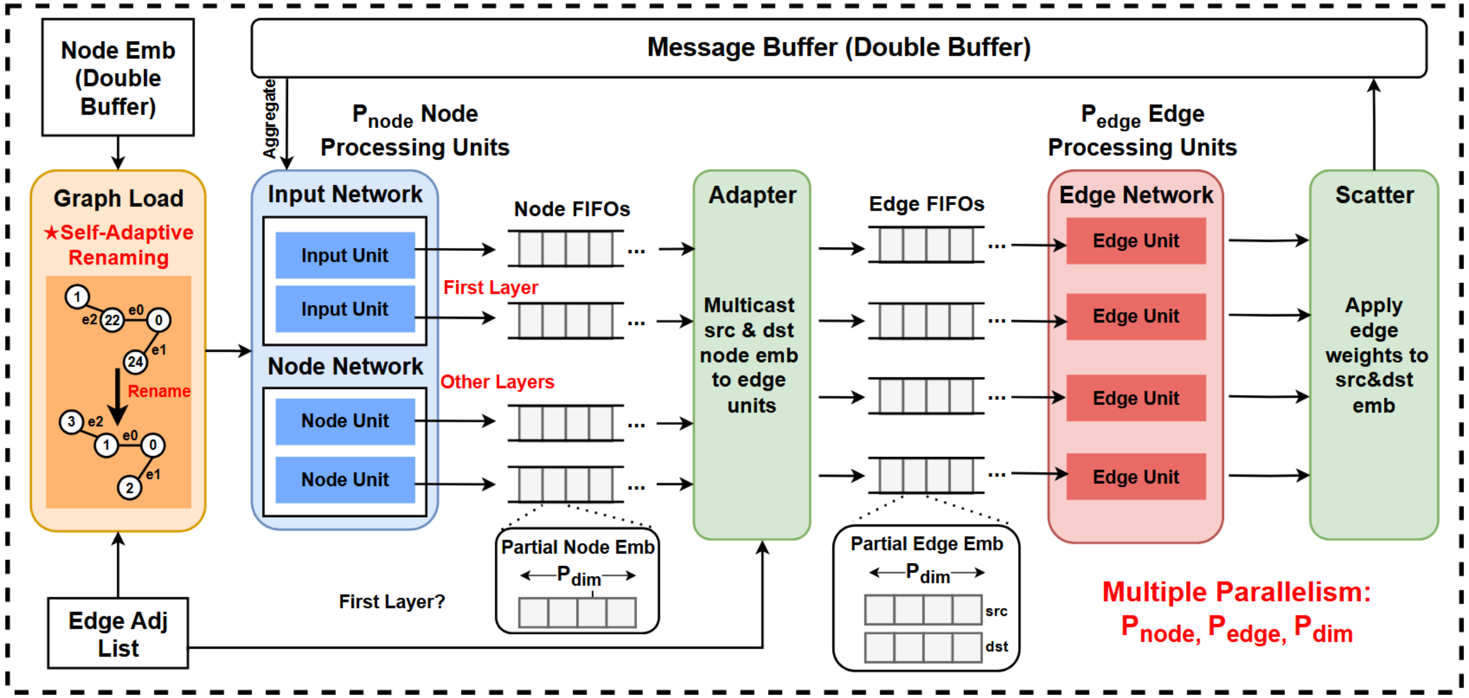
Approach 1: FlowGNN

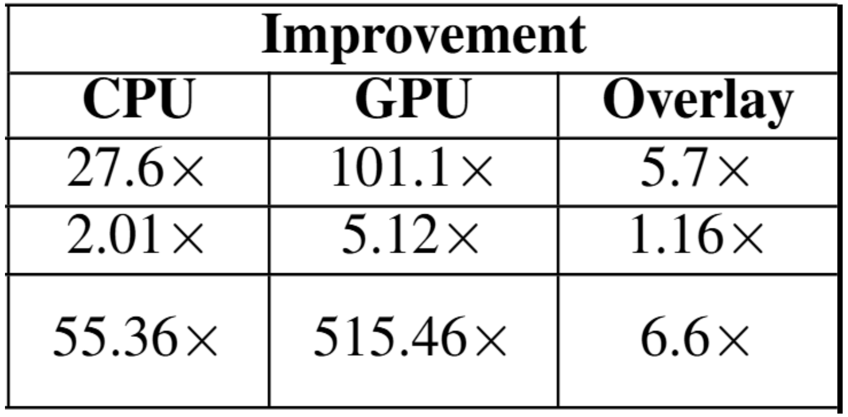
❑FlowGNN is a flexible architecture for Graph Neural Network acceleration on FPGAs

❑Manual implementations, from PyTorch → C → Verilog, using High Level Synthesis (HLS)

➢ Version 1: Track construction only:
  - **8.82 us per graph** (Freq. 285 MHz), tested with: 92 nodes, 142 edges

➢ Version 2: from Hits -> Clustering → Triggering:
  - **9.2 us per graph** (Freq. 180 MHz), Tested with: 92 nodes, 142 edges





Improvement		
CPU	GPU	Overlay
27.6×	101.1×	5.7×
2.01×	5.12×	1.16×
55.36×	515.46×	6.6×

Approach 2: *hls4ml*

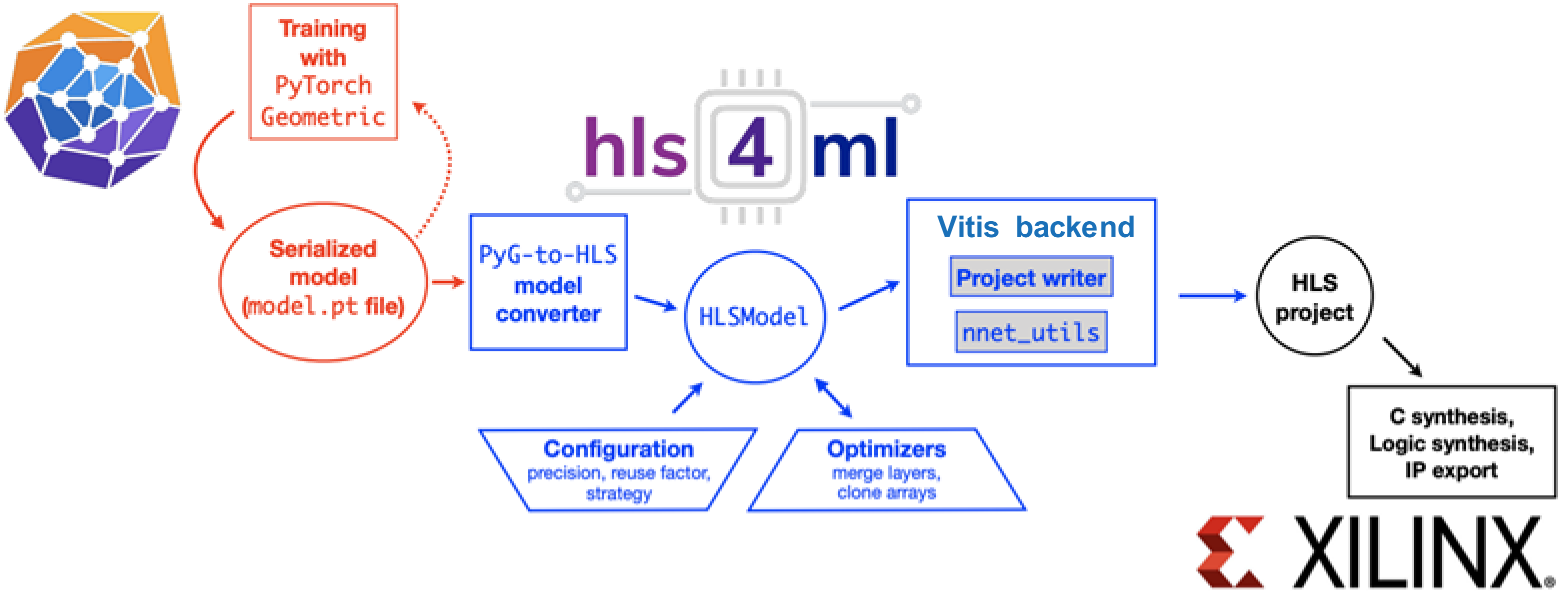
- hls4ml (arXiv: 1804.06913)

❑ ***hls4ml*** is a HEP community developed compiler taking Keras, Pytorch, or ONNX input and producing High Level Synthesis (HLS) code implementing the network as spatial dataflow.

  ➢ HLS code is usually C++ or similar with directives to guide the produced hardware.

  ➢ hls4ml has different "backends" for the different flavors of HLS desired by tools.

❑ GNN support is under development: currently the process is not as automated as for other network types, manually implemented a simpler model, hits -> trigger


Training with PyTorch Geometric
Serialized model (model.pt file)
hls 4 ml
PyG-to-HLS model converter
HLSModel
Configuration
precision, reuse factor, strategy
Optimizers
merge layers, clone arrays
Vitis backend
Project writer
nnet_utils
HLS project
C synthesis, Logic synthesis, IP export
XILINX

hls4ml Initial Implementation (MVTX-only MLP)

- **The MLP-layerwise model has been synthesized for the FPGA**
- The model consists of two parts
  - The first part, called the **aggregation step**, collects all the clusters. It is called for each cluster in a bunch crossing. This needs a high throughput: initiation interval every 1 clock cycle, 117 ns latency
  - The second part, called the **prediction step**, is called once per bunch crossing, to make a prediction based on the ingested clusters: 63 clock cycles, 308 ns latency
- The two models are synthesized separately, with the FPGA utilization for the FELIX 712 given below, using Vitis HLS and Vivado 2024.1.

	Aggregation step	Prediction step
LUT	23 587 (3.56%)	16 582 (2.50%)
FF	15 129 (1.14%)	31 226 (2.35%)
DSP	19 (0.34%)	498 (9.02%)
BRAM	0 (0%)	30.5 (1.41%)

Approach 3: Bipartite Model

✅ The overall latency is **reduced by 14.6%** on FPGA

⚠ However:

- ➢ BRAM usage increased a lot because of more parameters
- ➢ The clock frequency drops from 285 MHz → 198 MHz (30% reduction) due to heavier resource utilization
- ➢ More latency is spent on data movement for parameter loading

📝 Planned:

- ➢ More aggressive quantization, Model pruning if possible

GCN (previous results) @ 285 MHz				**Bipartite Model (new results) @ 198 MHz**			
DSP	**FF**	**BRAM**	**Latency**	**DSP**	**FF**	**BRAM**	**Latency**
488 (5.4%)	214K (8.2%)	406 (20.2%)	**8.82 us / graph**	359 (3.9%)	344K (13.2%)	1766 (87.6%)	**7.53 us / graph**
							14.6% ↓

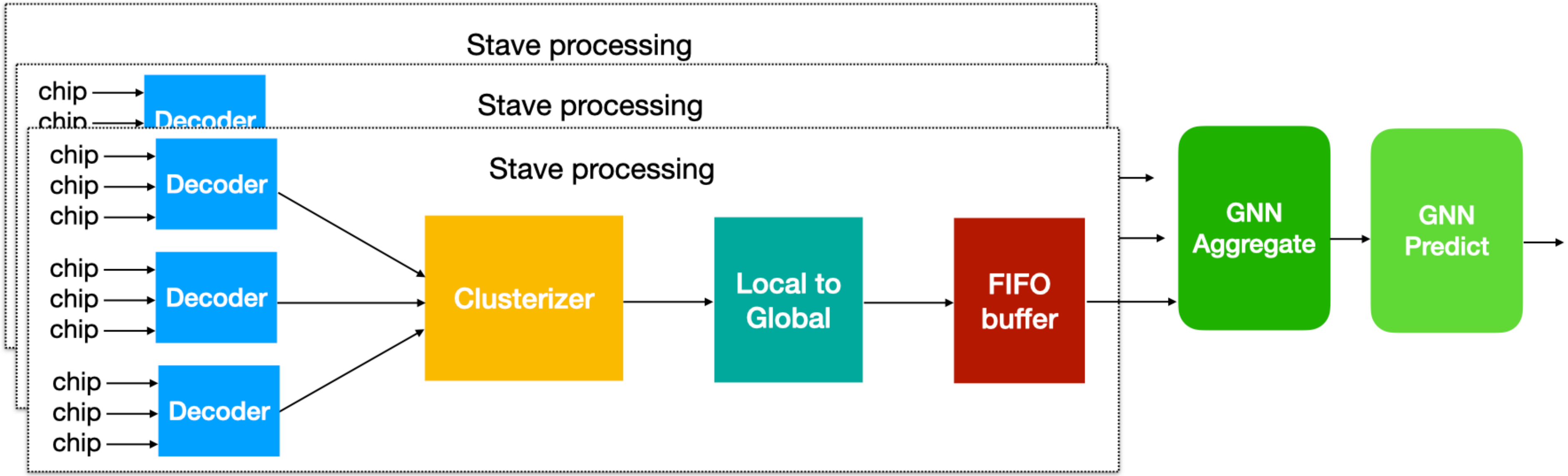
Demonstrator Implementation

- Jovan Mitrevski (FNAL)
- Jo Schambach (ORNL)

Demonstrator Implementation

- ❑ Designed firmware for FLX-712 AI board with simple MVTX-only processing
- ❑ Each stave processing unit has:
  - ➢ Three decoders, each decoding the the input of three pixel chips
  - ➢ One clusterizer grouping adjacent energy deposits into clusters.
  - ➢ Local to global converter
- ❑ Simple MLP layerwise GNN, one per board

FELIX-712



Smaller Scale Demonstrator:

- with MVTX Telescope Communication

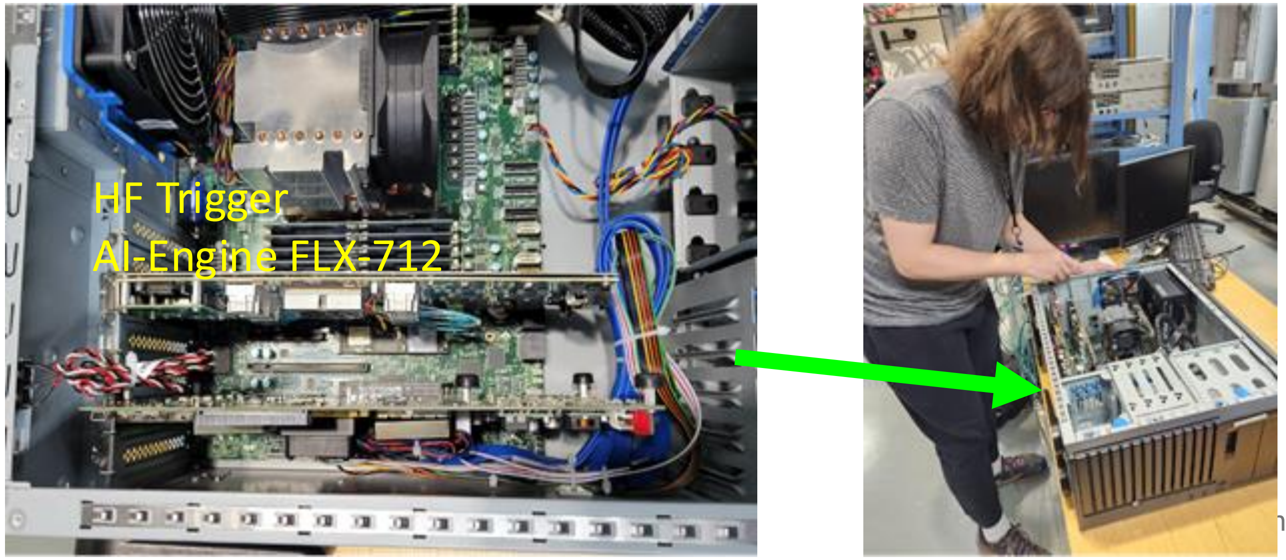
Due to very tight sPHENIX operation schedule and certain detector challenges, we didn't get the opportunity to integrate AI/ML system into the sPHENIX DAQ, instead, used MVTX telescope in the sPHENIX counting house for the system test

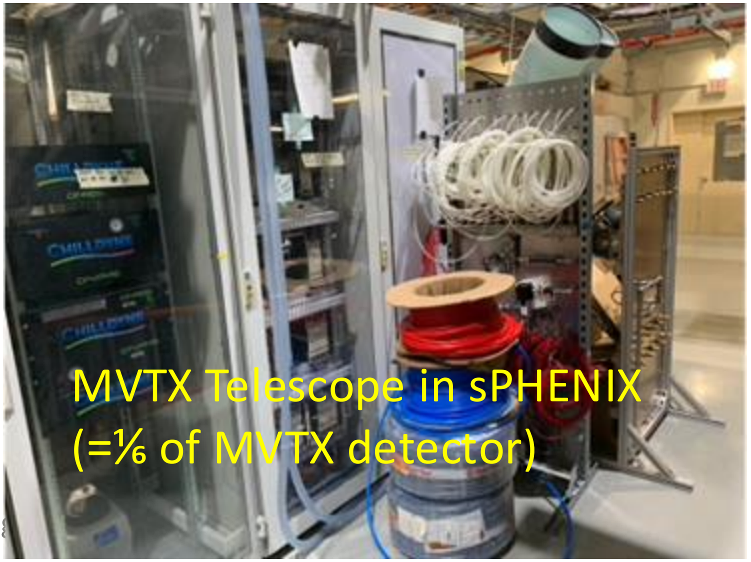
❑ FELIX-712 was designed as sPHENIX readout board, the PCIe is used to receive data from the optics

- ➢ Save the timing (Bunch Crossing ID) and trigger decision from the AI
- ➢ Configured the PCIe uplink (normally used just for configuration) to load real detector data to the board, for a controlled validation environment

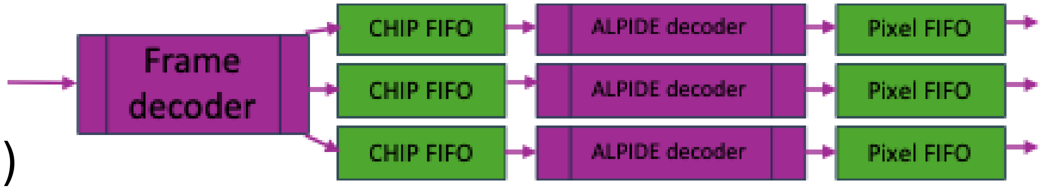
❑ Successfully received and decoded data from single stave of the MVTX 8-stave telescope (MVTX = 6 x Telescope)

❑ Added ILA via Xilinx virtual cable for additional debugging and monitoring





MVTX Decoder Development (Conventional)



- ❑ **First FPGA-based decoder for ALPIDE sensors**
  - ➢ The design has been simplified
    - There is only one set of buffers (instead of per event)
  - ➢ The design was validated on simulation, PCIe and Telescope data
    - This also helped to validate the PCIe and Telescope comms
  - ➢ Due to MVTX data compression we need 1 decoder module per detector (FeeID) link (144 total)
    - Table below is from 1-stave implementation, except for last 2 lines

	LUT (663K)	FF (1.3M)	BRAM (2K)
Frame decoder	151	287	0
ALPIDE decoder (x3)	343	256	0
FIFOs (x6)	31	36	1
Total per FeeID	1366	1271	6
Total per half- barrel (est. from 1-stave, 2024)	**98K (14.7%)**	**91K (7%)**	**432 (21%)**
Total per half- barrel (24-stave implem.,2025)	**137K (20%)**	**94K (7%)**	**432 (21%)**

Implementing Multiple Staves

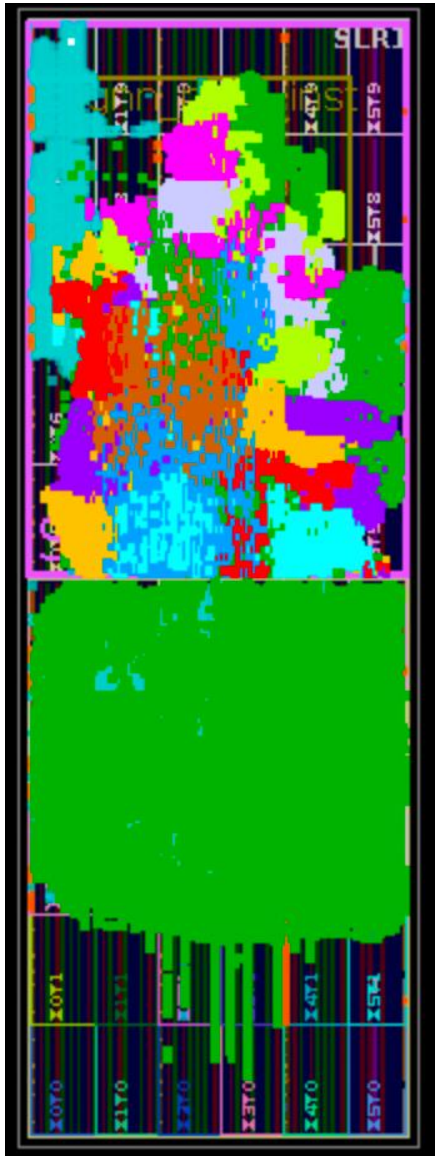
- ❑Implementing 24 staves in the FLX-712 proved impossible:
  - ➢timing failed badly due to congestion.
  - ➢A larger FPGA, like a FLX-155 is needed (considered for EIC readout)
- ❑Meeting timing with 8 staves was also complicated
  - ➢Needed to reduce the PCIe size: go to one endpoint instead of the default 2–this should fine for the AI board since the PCIe is only for monitoring and control
  - ➢Needed to use some floor-planning to optimize the synthesis and implementation strategies

Note:

- MVTX = 48 staves, 6 FLX-712 cards
- 8 staves per FLX-712

FPGA Resource Utilization (FLX-712)

green: PCIe
other colors: staves 0-7
brown: hsl4ml aggregate
blue: hsl4ml predict



- Currently we have both one-stave and eight-stave implementations:
  - Note: 8 stave has only one PCIe endpoint, 1-stave has two
- Also include 24-stave version, one PCIe endpoint numbers, which badly misses timing

	LUT (663K)	FF (1.3M)	BRAM (2K)	DSP (5.5K)
1-stave	170K (25.7%)	361K (27.2%)	1.0K (47%)	536 (9.7%)
8-staves	216K (32.5%)	282K (21.3%)	0.7K (33%)	669 (12.1%)
24-stave	398K (60.0%)	422K (31.8%)	1.2K (54%)	1033 (18.7%)

- Ideally, for lower latency, one would have one clusterizer per decoder, instead of per 3 decoders.

	LUT (663K)	FF (1.3M)	BRAM (2K)	DSP (5.5K)
clusterizer	4.2K (0.6%)	3.0K (0.2%)	0	0
8-staves (24 clus)	283K (43%)	329K (25%)	0.7K (33%)	669 (12.1%)
24-stave (72 clus)	600K (90%)	564K (43%)	1.2K (54%)	1033 (18.7%)

Note: New FLX-155 ~ 3x FLX-712 in resources

FPGA Ready Algorithm Summary: It is fully feasible!

- ❑ Including INTT would significantly improve performance
  - ➢ Additional decoding, clusterizing, and event-building logic
  - ➢ Nevertheless, 8-stave implementation has some room for further improvements
- ❑ Currently implementing a simple GNN: 7% LUT, 9% DSP utilization
  - ➢ Adding INTT would increase these
  - ➢ There is room to include a more performant model
- ❑ Nevertheless, FLX-712 is quite old. Applications at, for example, the EIC would use newer FPGAs (such as FLX-155), which would:
  - ➢ Allow more complicated/avanced AI models
  - ➢ Cover larger sector per board

EIC Challenges

- Cameron Dean (MIT)

- Yasser Corrales Morales (LANL, MIT)

EIC – be prepared for unexpected

- lessons learned from sPHENIX data taking and implications for future EIC and other experiments

EIC
A NEW ERA OF DISCOVERY
THE 2023 LONG RANGE PLAN FOR NUCLEAR SCIENCE
A NEW ERA OF DISCOVERY
THE 2023 LONG RANGE PLAN FOR NUCLEAR SCIENCE

New ideas being developed
to address unexpected challenges …

Unexpected Challenges First Observed in sPHENIX 2023 Au+Au Runs

- Full streaming readout in high beam backgrounds!

❑ **Major beam-related background with Au beam**

➢ Related to beam halo induced particles hitting large number of sensor pixels in the MVTX detector sensors

NO problem in p+p collisions

Data rate >> DAQ bandwidth! (>10^3)

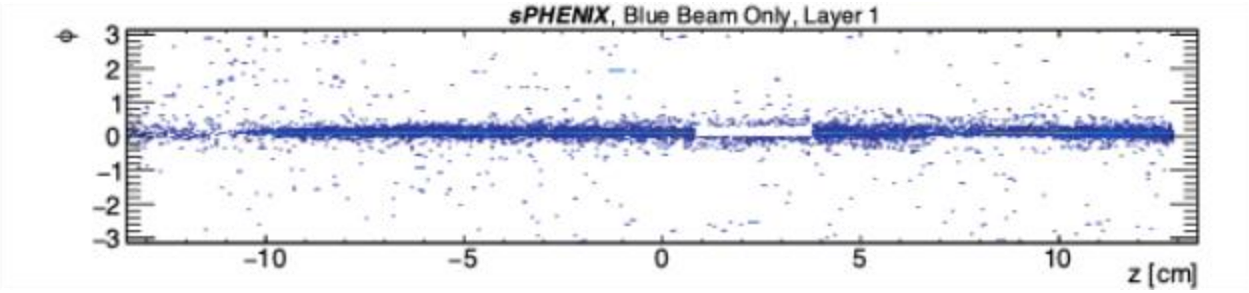
EIC: day-1, e+A program

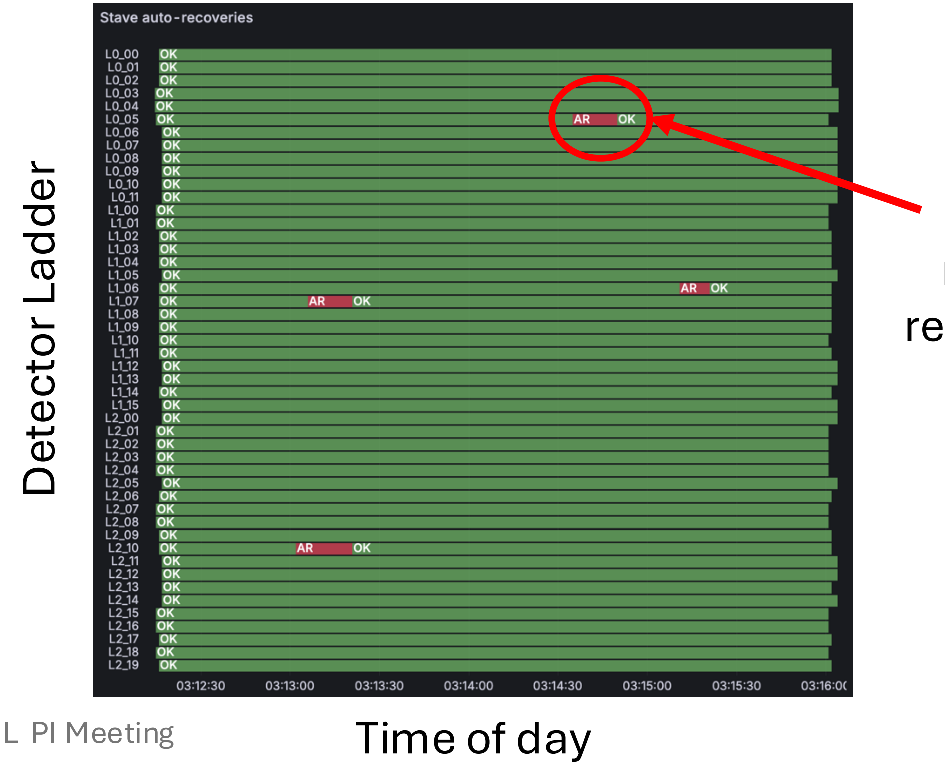
- Could face similar high backgrounds with ion beams

Smart data management highly desired on/near the detectors for full streaming readout in high background environment would enable masking of beam backgrounds from detectors before a reset is necessary

Expected hits: O(10s) out of ~1M pixels/stave

Beam halo background hits: >> O(10k)



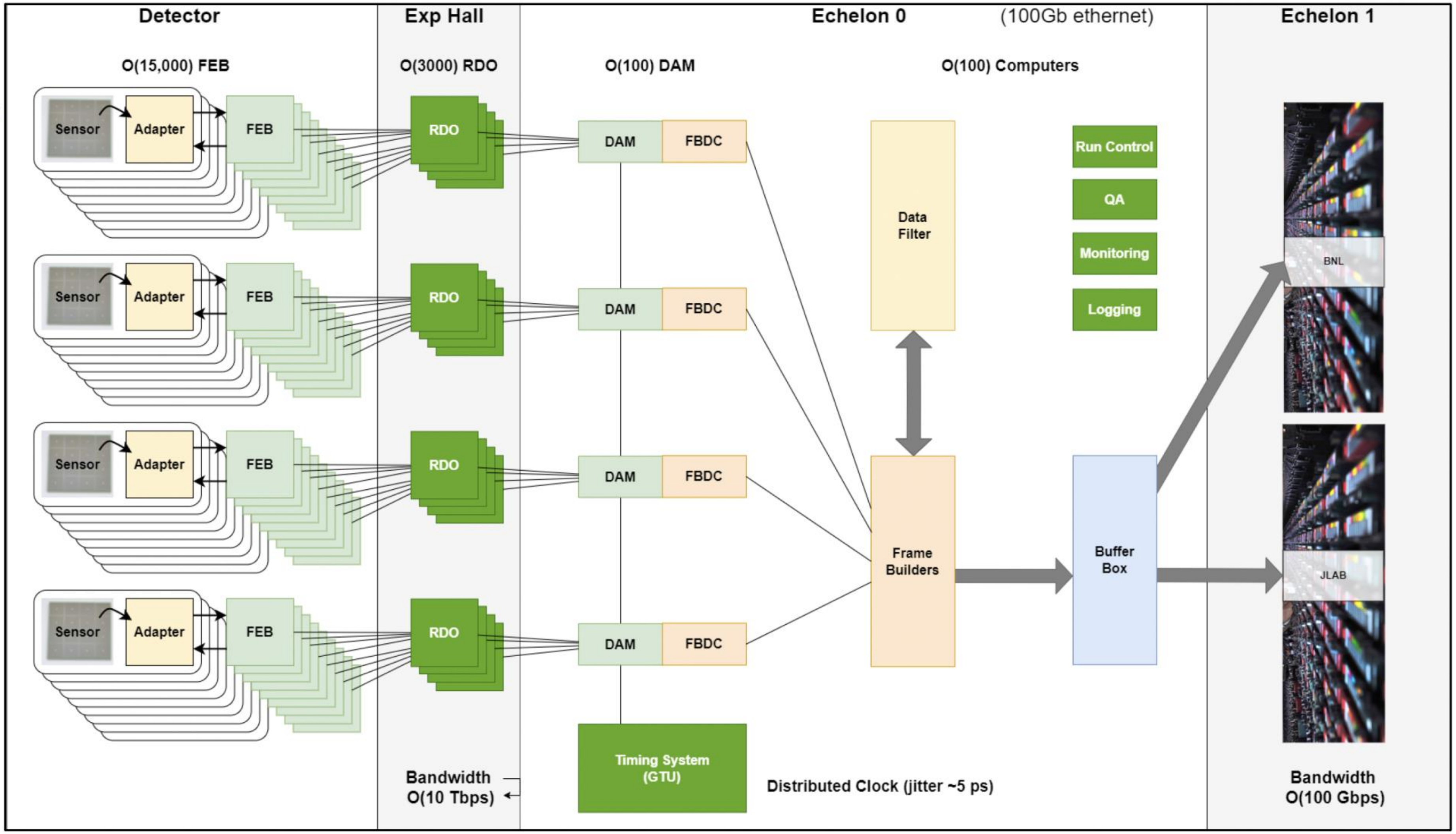


Stave auto-recoveries
L0_00 L0_01 L0_02 L0_03 L0_04 L0_05 L0_06 L0_07 L0_08 L0_09 L0_10 L0_11
L1_00 L1_01 L1_02 L1_03 L1_04 L1_05 L1_06 L1_07 L1_08 L1_09 L1_10 L1_11 L1_12 L1_13 L1_14 L1_15
L2_00 L2_01 L2_02 L2_03 L2_04 L2_05 L2_06 L2_07 L2_08 L2_09 L2_10 L2_11 L2_12 L2_13 L2_14 L2_15 L2_16 L2_17 L2_18 L2_19
OK
AR
03:12:30
03:13:00
03:13:30
03:14:00
03:14:30
03:15:00
03:15:30
03:16:0(
Detector Ladder
Time of day

Beam background interaction requiring 20s reset of detector area

EIC SRO … the data throughput challenge

- Bunch Crossing
  ~10.2 ns/98.5 MHz
- Interaction Rate
  ~ 2 us/500 kHz
- Low occupancy

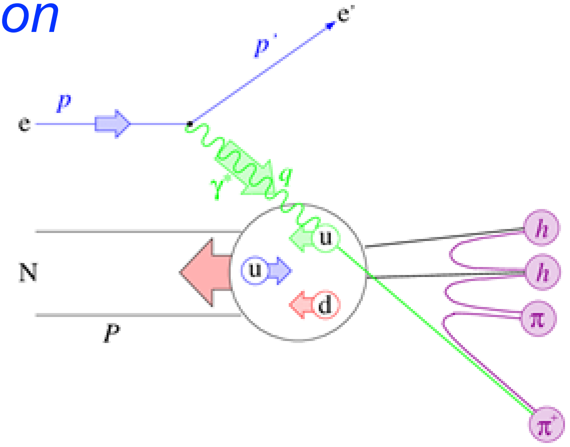


A big unknown: beam backgrounds, could easily overwhelm the DAQ system!

Better be prepared~

Fast-ML for EIC – work in progress…

- DIS-electron identification in real-time with beam background suppression



Selective streaming readout for AI-Engine:

❑ tag DIS-electron to define DIS event ID

➢ EMCal + Trker + ePID

➢ DCA~0

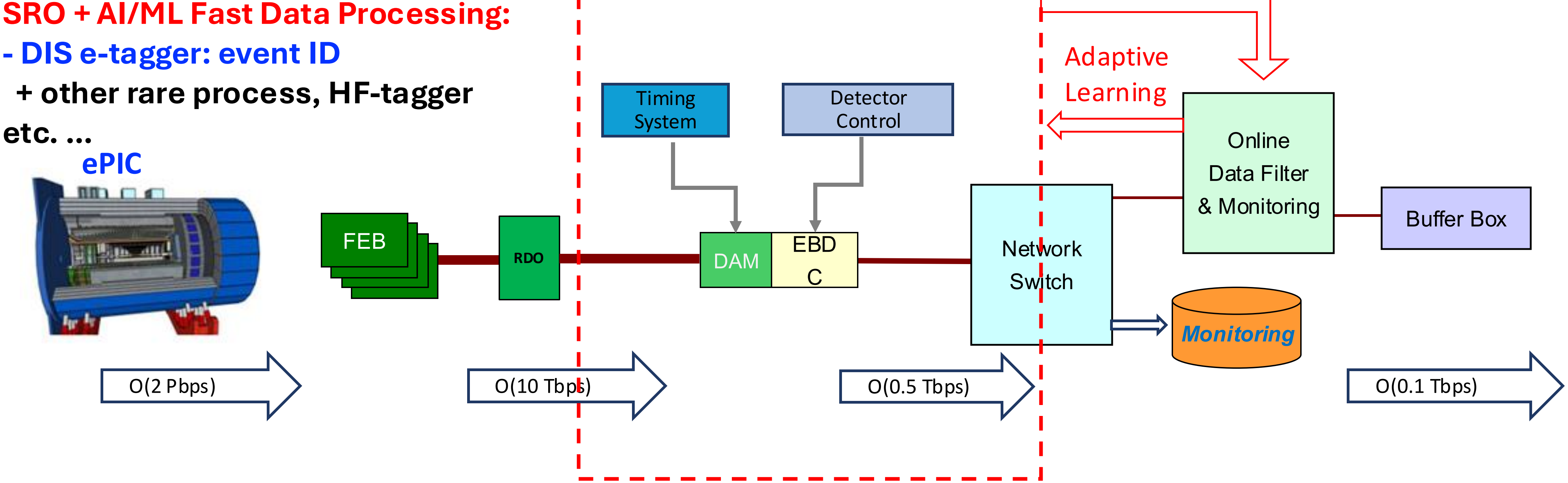
❑ With AI noise suppression on chip (AI-on-Sensor)!

Add AI-based active beam halo background rejection: AI-on-Detector!

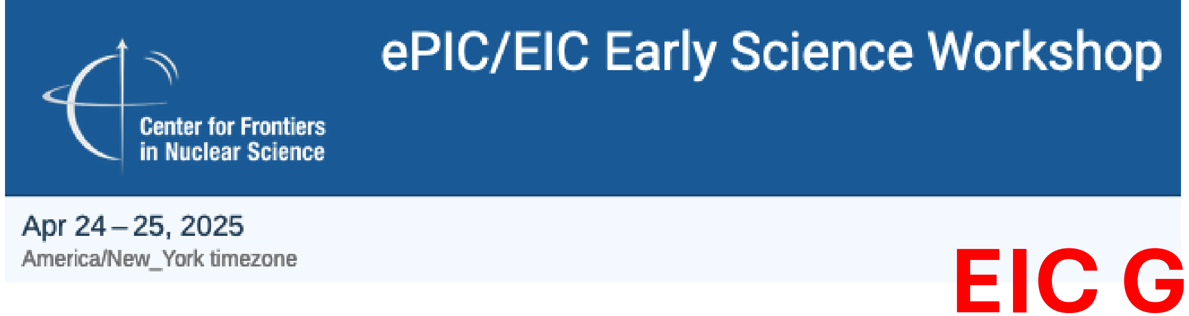
SRO + AI/ML Fast Data Processing:

- DIS e-tagger: event ID

+ other rare process, HF-tagger etc. …



Backup slides



ePIC/EIC Early Science Workshop

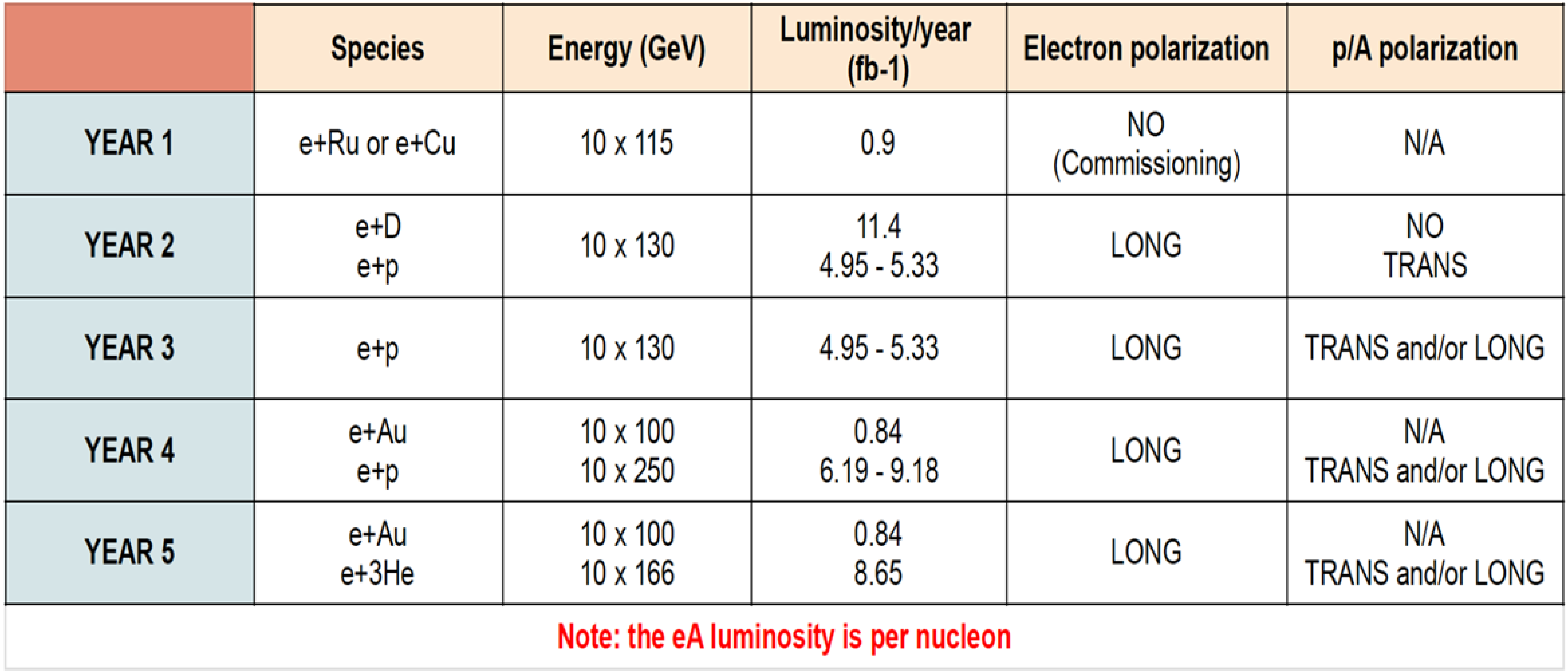
Center for Frontiers in Nuclear Science

Apr 24 – 25, 2025

America/New_York timezone

https://indico.cfnssbu.physics.sunysb.edu/event/410/timetable/#20250424

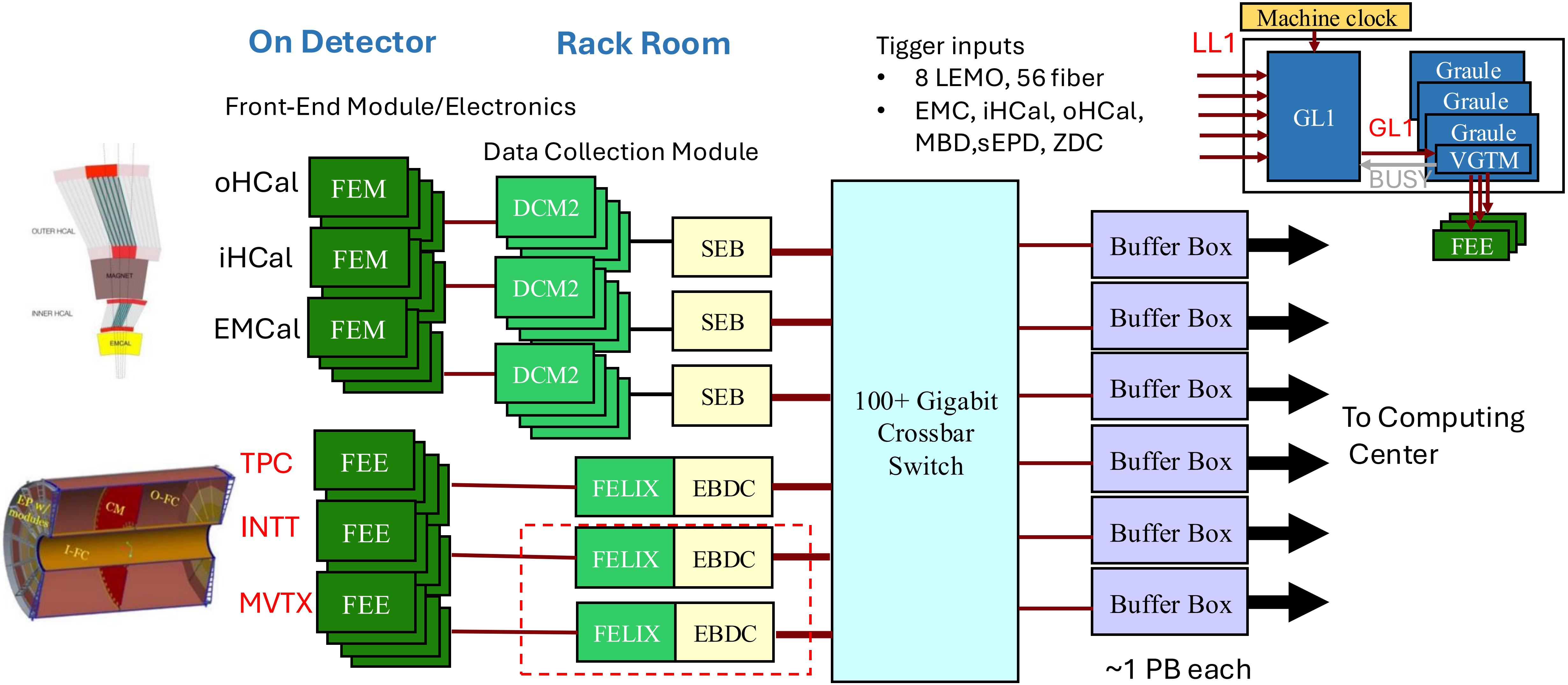
EIC Game Plan



	Species	Energy (GeV)	Luminosity/year (fb-1)	Electron polarization	p/A polarization
YEAR 1	e+Ru or e+Cu	10 x 115	0.9	NO (Commissioning)	N/A
YEAR 2	e+D e+p	10 x 130	11.4 4.95 - 5.33	LONG	NO TRANS
YEAR 3	e+p	10 x 130	4.95 - 5.33	LONG	TRANS and/or LONG
YEAR 4	e+Au e+p	10 x 100 10 x 250	0.84 6.19 - 9.18	LONG	N/A TRANS and/or LONG
YEAR 5	e+Au e+3He	10 x 100 10 x 166	0.84 8.65	LONG	N/A TRANS and/or LONG

Note: the eA luminosity is per nucleon

sPHENIX Readout and Trigger Distribution



On Detector

Rack Room

Front-End Module/Electronics

Data Collection Module

Tigger inputs
- 8 LEMO, 56 fiber
- EMC, iHCal, oHCal, MBD,sEPD, ZDC

Machine clock

LL1

GL1

GL1

Graule

Graule

Graule

VGTM

BUSY

FEE

oHCal

iHCal

EMCal

FEM

FEM

FEM

DCM2

DCM2

DCM2

SEB

SEB

SEB

TPC

INTT

MVTX

FEE

FEE

FEE

FELIX

EBDC

FELIX

EBDC

FELIX

EBDC

100+ Gigabit Crossbar Switch

Buffer Box

Buffer Box

Buffer Box

Buffer Box

Buffer Box

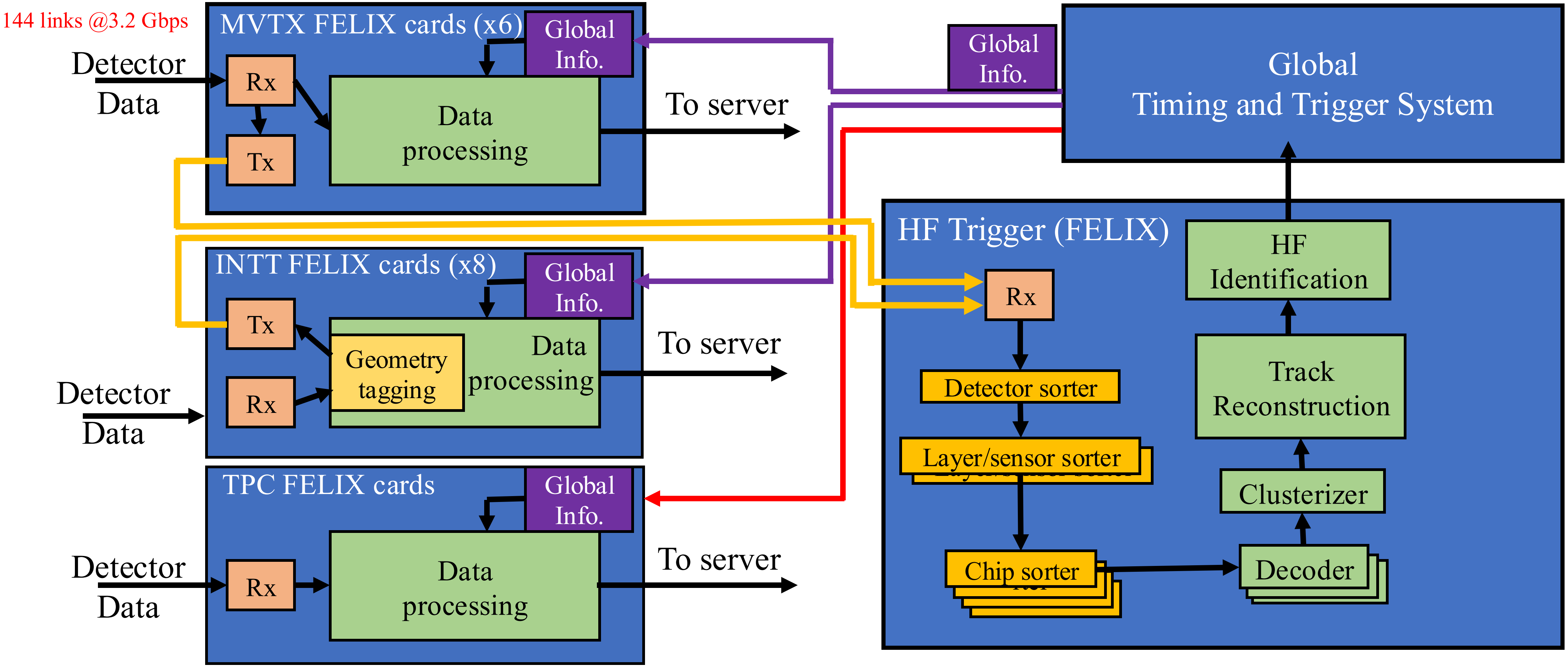
Buffer Box

To Computing Center

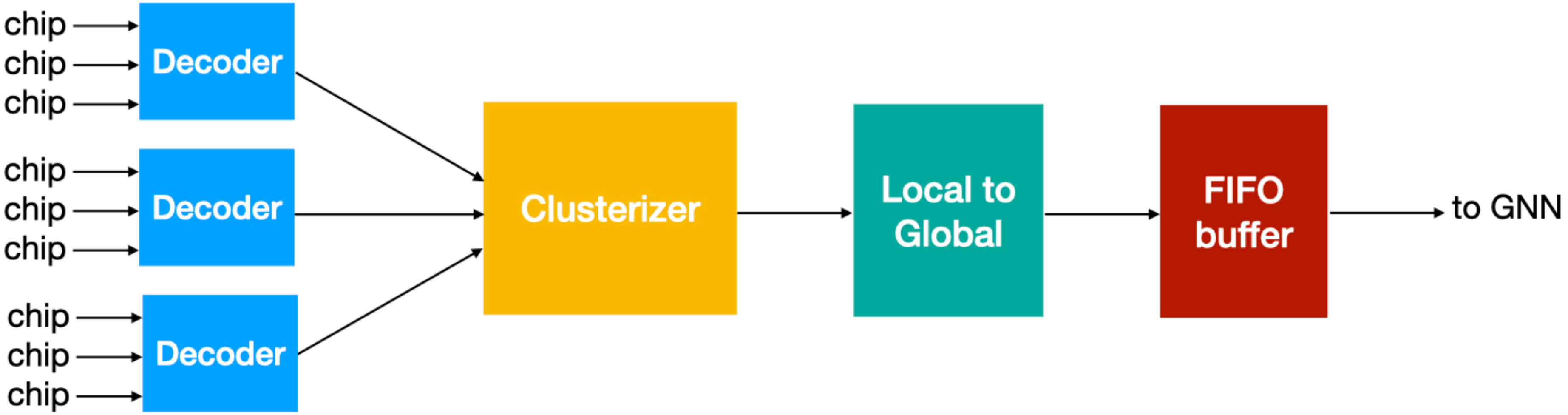
~1 PB each

SRO(MVTX+INTT) -> ML -> HF Trigger

HF Trigger System Diagram



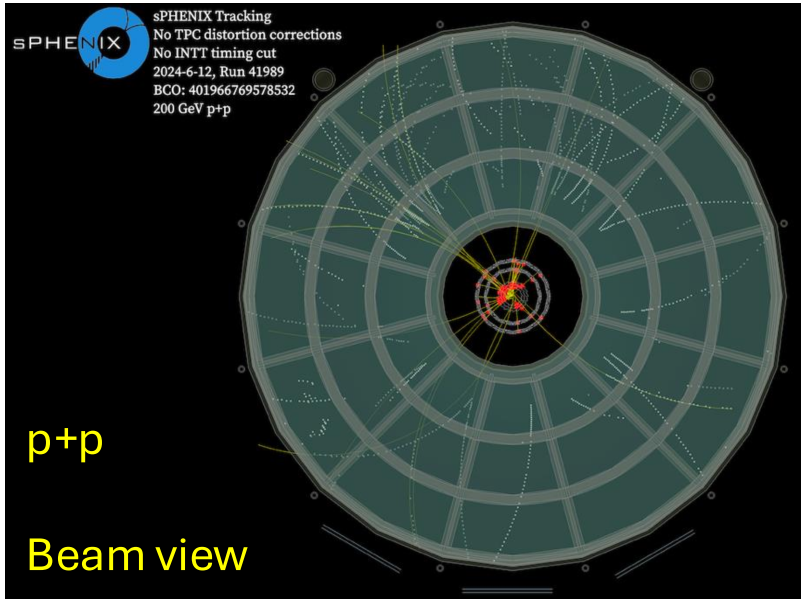
MVTX Stave Processing

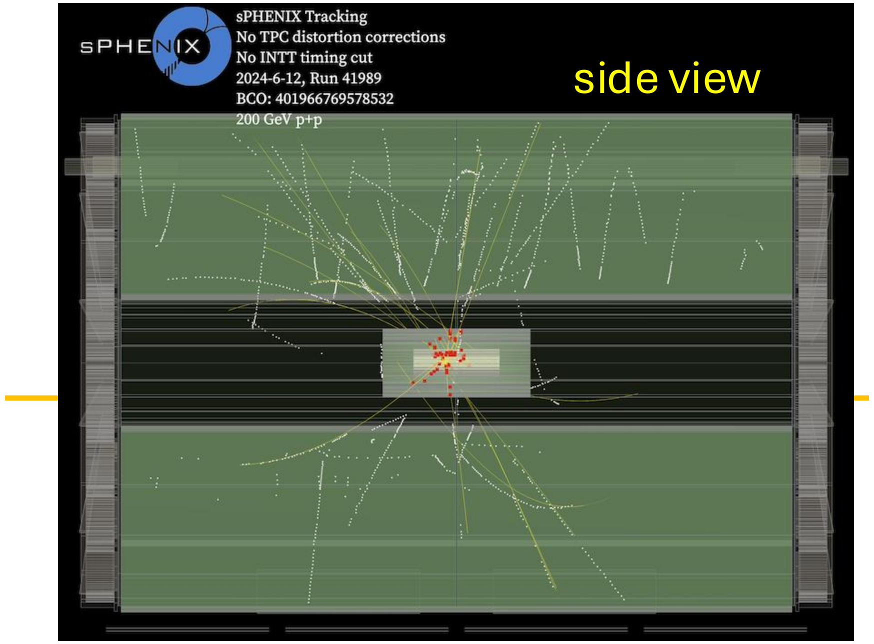


- The diagram above shows how a single stave is processed
  - ➢ 8 or 24 staves duplication implemented, 24 failed badly with FLX-712, 8 barely fit …
  - ➢ should fit with latest FPGA
- The clusterizer groups adjacent energy deposits into clusters.
- One GNN instance of the MLP-layerwise placed on each board
  - ➢ The GNN aggregator takes the data from one event out of each stave in sequence before sending it to the prediction step

AI/ML Algorithm Development

- □ An efficient, end-to-end, robust trigger pipeline capable of handling multi-collision pileup
  - ➢ pileup of p+p collisions: hits from ~20 events
- □ Two stages of pipeline:
  - ➢ Stage 1: Tracking
    - Connect hits left by the same particle to create tracks
    - Reduce data size by eliminating hits left by pileup events
  - ➢ Stage 2: Trigger decision
    - Given tracks, predict whether the event is a HF event
- □ Developed algorithm NOT sensitive to the IP variations
- □ Improve performance by reinforcing physics laws in the models

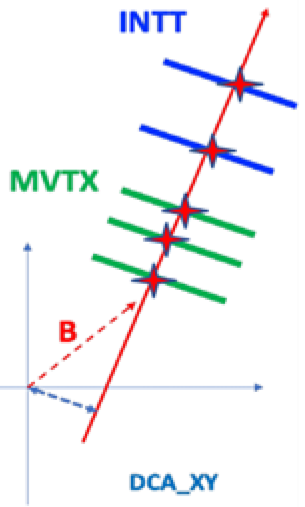


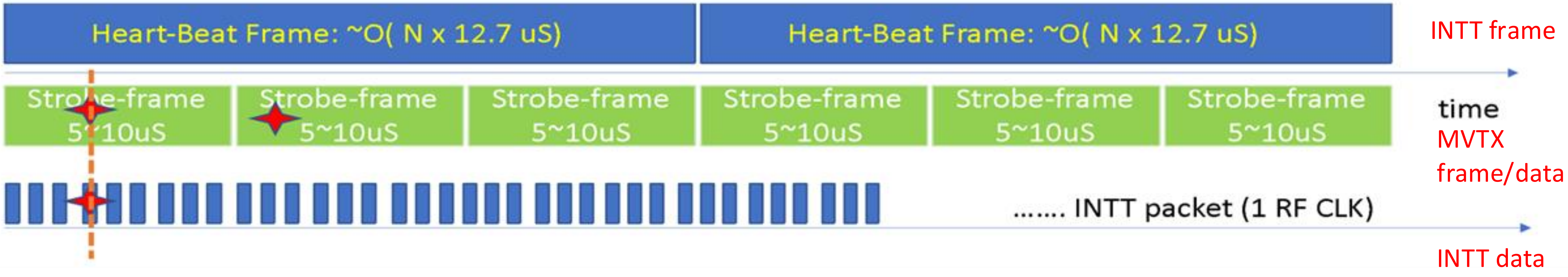


Raw Data Pre-processing: Event Building

A big challenge of data integration!

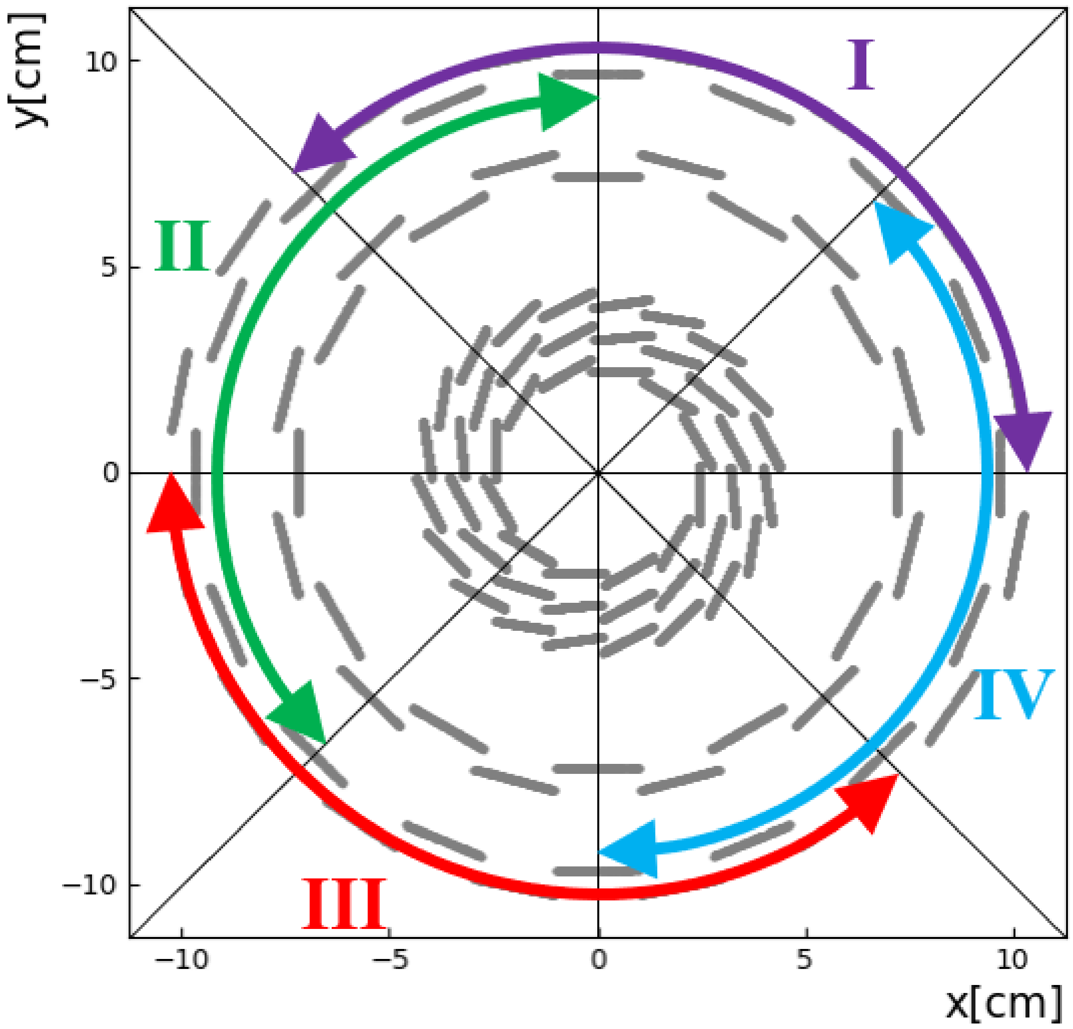
- ❑ With the current MVTX-only setup the event building is easy
  - ➢ Since the detector links contain Bunch Crossing ID we can just read event by event link by link
- ❑ Challenge: once we add INTT stream this will be much more complicated due to different reading stream lengths and latencies
- ❑ Important is to first have the simpler MVTX-only implementation working!



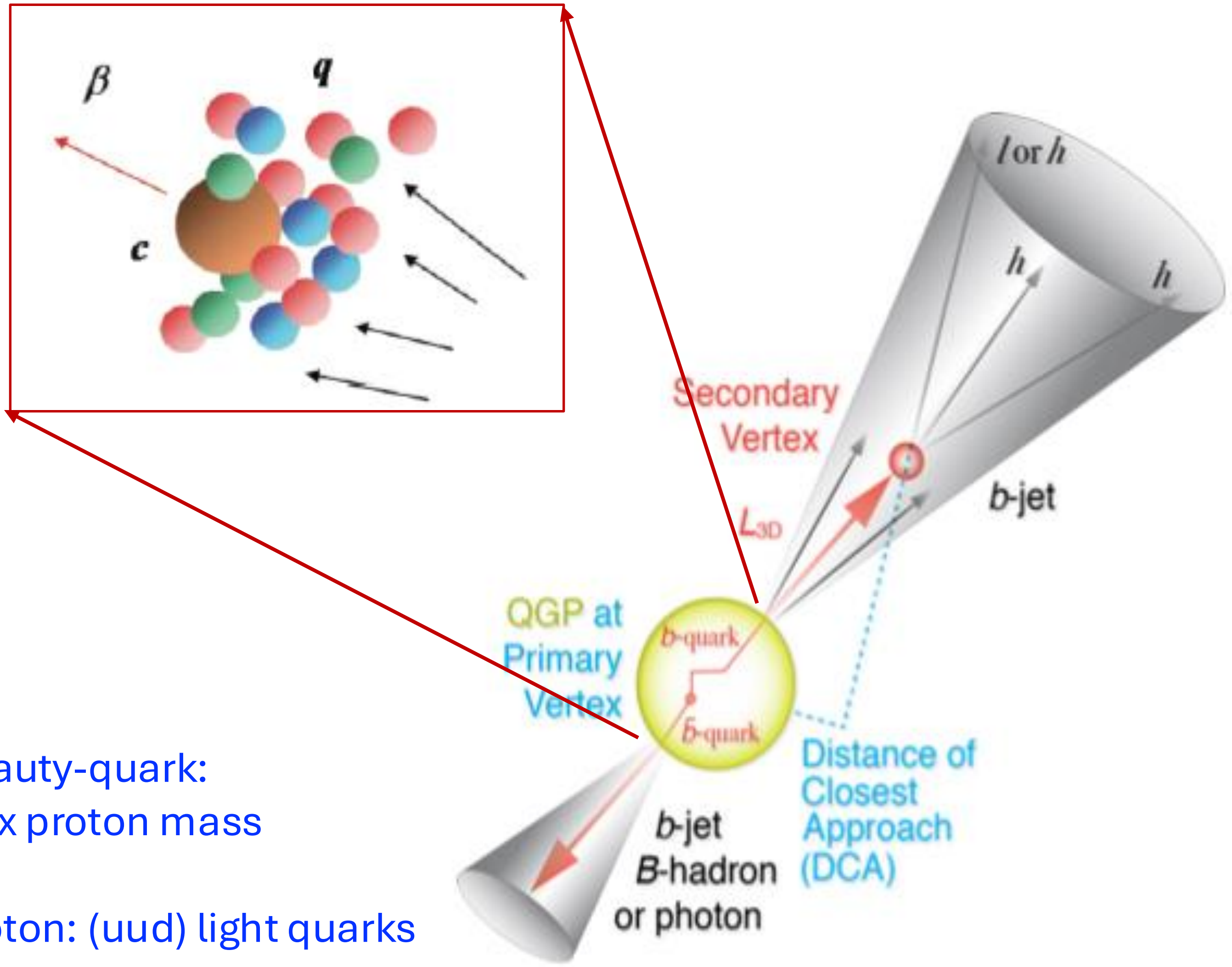


Alternative – more Partitions for Parallel Processing

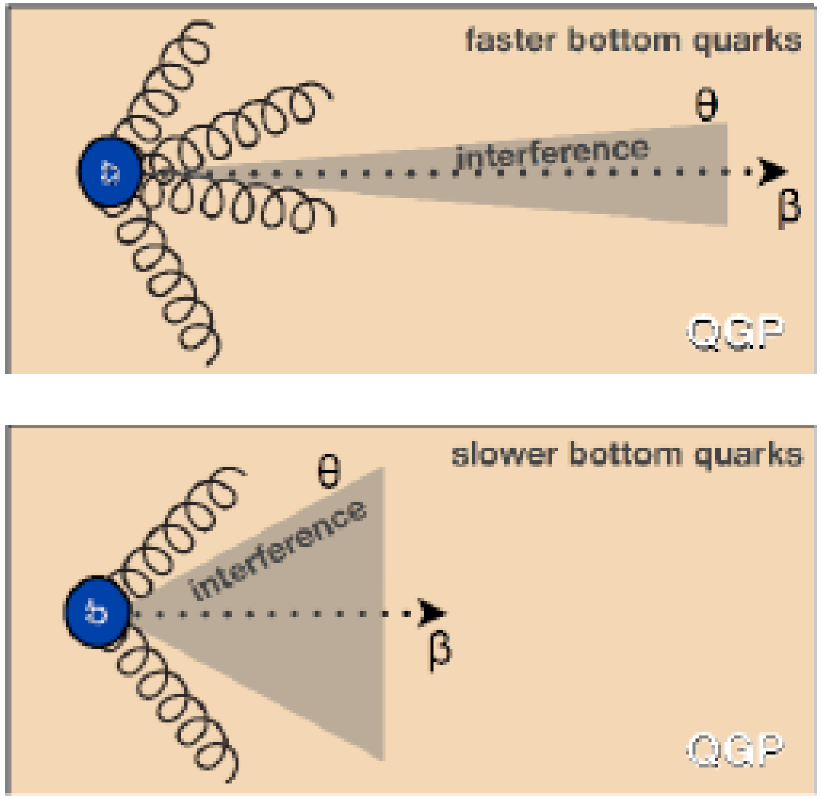
- ❑ **8 sectors** evenly divided along the azimuth angle φ
- ❑ **3** consecutive sectors form a **Zone**
- ❑ Adjacent zones share one overlapping sector
- ❑ Data streams within each zone are processed in parallel



Heavy Quark Physics: a Pilar of RHIC Science



B-quark radiative energy loss in QGP
- Less dE/dx due to heavy mass



Beauty-quark:
~ 5x proton mass

Proton: (uud) light quarks