EPSCI

Experimental **P**hysics **S**oftware and **C**omputing **I**nfrastructure

David Lawrence - EPSCI Group Lead

June 12, 2024

EPSCI

Experimental Physics Software and Computing Infrastructure

EPSCI
Experimental Physics Software and Computing Infrastructure

Mission Statement:

Identify, develop, implement, and maintain software and computing technologies in support of the Jefferson Lab Science Program.

EPSCI Priorities:

1. Support Experimental arm of the Jefferson Lab 12 GeV and EIC Science Program through maintenance of existing software.

2. Investigate and develop new software and computing technologies to aid the 12 GeV and EIC science program at Jefferson Lab.

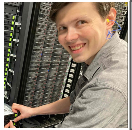
3. Promote the software and computing technologies developed at Jefferson Lab to benefit the science programs throughout the national lab system.

EPSCI Members

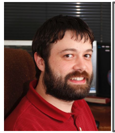
group formed Feb. 2020

Group Lead

David Lawrence, PhD (physics)
Expertise: Physics, C++, software framework, online systems



Nathan Brei, BS (aerospace engineering) MS (CS)
Expertise: Programming languages, parallel processing



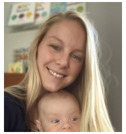
Thomas Britton, PhD (physics)
Expertise: Physics, software, OSG, AI DQM

Michael Goodrich, MS (physics) PhD (Computational Modeling/Simulation)
Expertise: modeling, sim., sci comp, DS / ML, physics

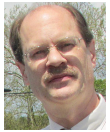
Vardan Gyurjyan, PhD (physics) MS (CS)
Expertise: Data Acquisition, Java, C++, software frameworks

Data Science Dept. 50% for AIOP

Armen Kasparian, MS (CpE)
Expertise: Data Science, Reinforcement Learning



Torri Jeske, PhD (physics)
Expertise: Experimental Nuclear Physics, Data Analysis, Detector Calibration



Carl Timmer, PhD (physics)
Expertise: Data Acquisition, Java, file format, I/O

postdoc

Jeng-Yuan Tsai, PhD (physics)
Expertise: Data Science, HTC Comp., programming

postdoc

Ayan Roy, PhD (CS)
Expertise: C++, Java programming

starting March 2024

Raiqa Rasool, BSCI (computer & information sciences)
Expertise: C++ programming, Web developer

HPDF Group 25% FTE for EPSCI

"Cissie" Xinxin Mei, PhD (CS)
Expertise: GPU, HPC systems

Information Resources Group 20% FTE for EPSCI

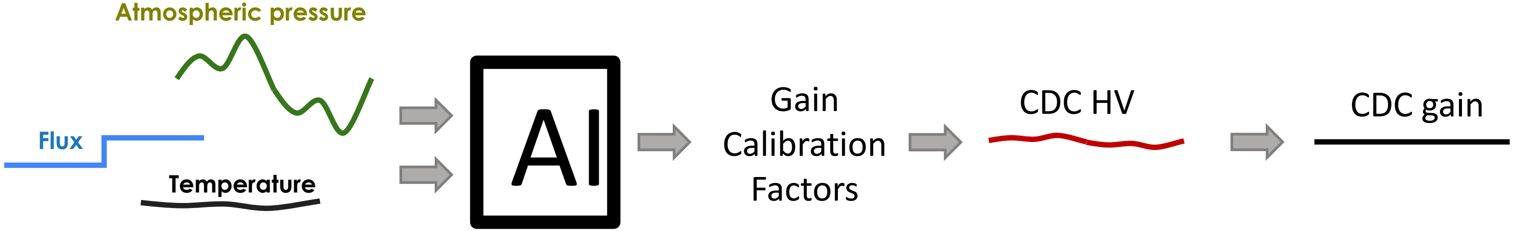
Nataliia Matsiuk, PhD (economics)
Expertise: Hydra Containerization

AIEC - AI for Experimental Controls

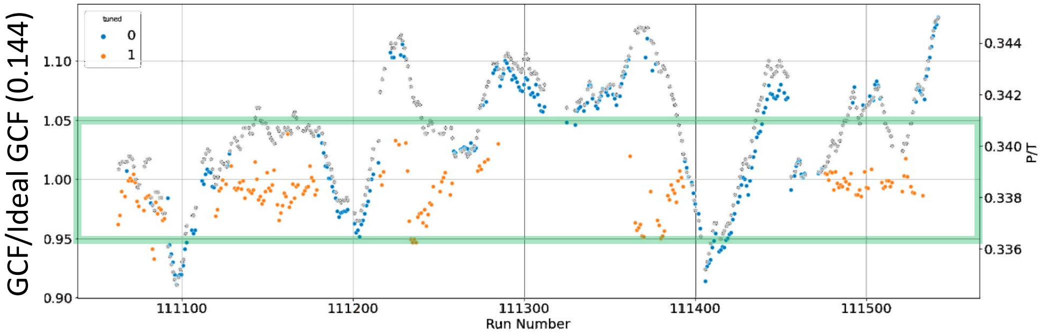
EXPERIMENTAL PHYSICS SOFTWARE AND COMPUTING INFRASTRUCTURE

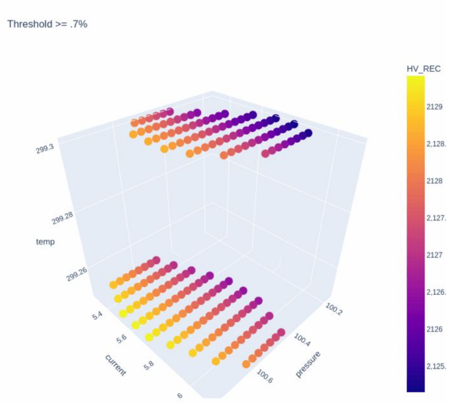
Developed system that uses AI/ML to determine control settings that are automatically applied during production data taking to stabilize gains of drift chambers.

AI trained on existing calibrations derived from data. ***Predicts calibrations from environmental values known prior to data taking.***



Deployment 3 – PrimEx-η June-Dec 2022





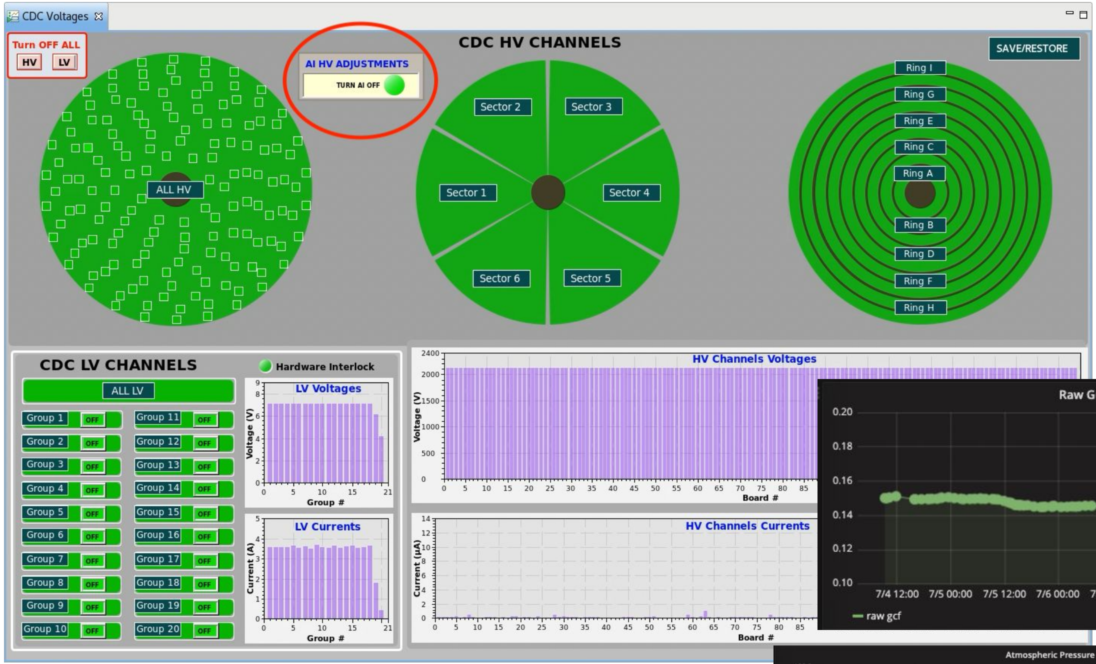
final policy: *when outside the volume of confidence revert to observation mode in order to gather more training data*

DOE NP LAB-20-2261
PI: David Lawrence
Co-PIs: Thomas Britton, Naomi Jarvis

AIEC - AI for Experimental Controls

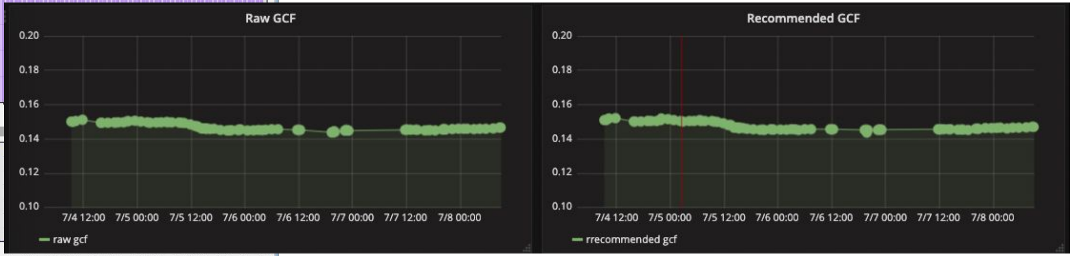
Integrating AI/ML into Standard Operations

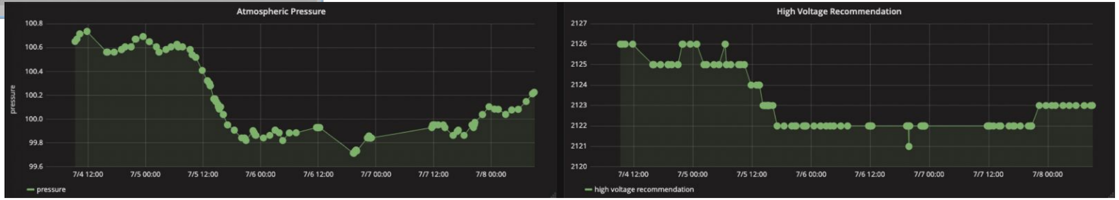
Experimental Physics Software and Computing Infrastructure



A switch was added to CDC Control GUI to allow shift takers to disable the AI/ML control completely.

Monitoring of the entire system was put onto a Grafana server.



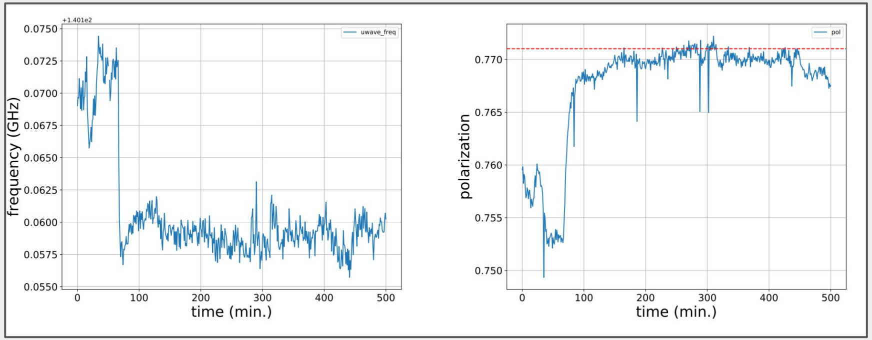


DOE NP LAB-20-2261
PI: David Lawrence
Co-PIs: Thomas Britton, Naomi Jarvis

Jefferson Lab
Thomas Jefferson National Accelerator Facility

AIOP - AI Optimized Polarization

Left: microwave frequency vs. time. Right: corresponding polarization

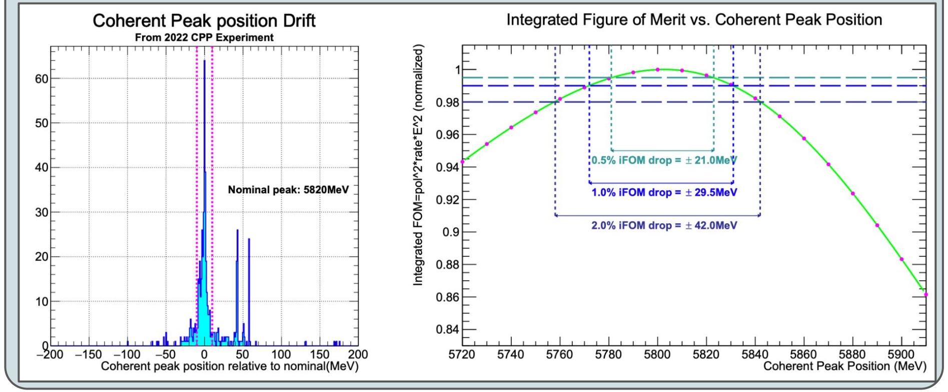


Polarized targets use microwave drivers to maintain polarization and NMR to measure it.
We will use AI/ML to:

- cleanly extract NMR signal from changing background
- adjust μ-wave frequency periodically to maintain optimal polarization

Polarized photon beam uses thin diamond radiator with precise angle adjustment to align edge of coherent bremsstrahlung peak.
We will use AI/ML to:

- determine angular shifts needed to maintain coherent peak of polarized bremsstrahlung photons within +/-10MeV its nominal position in real time
- digital-twin of diamond to map degradation as function of position in order to predict location of optimal polarization

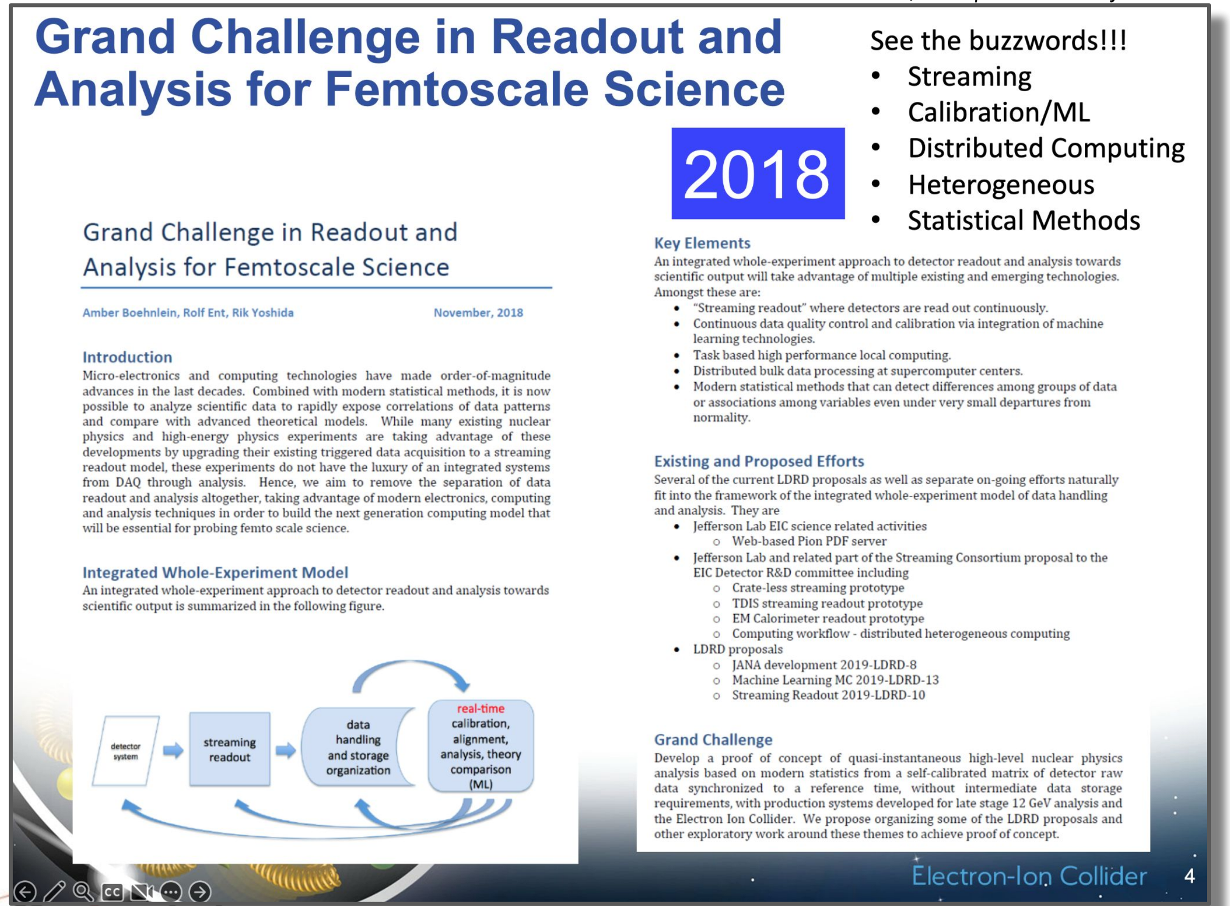


Funded through DOE NP FOA SC-0002875
PI: David Lawrence

Streaming Grand Challenge

EPSCI
Experimental Physics Software and Computing Infrastructure

slide from Jan. 5, 2024 presentation by Rolf Ent



Grand Challenge in Readout and Analysis for Femtoscale Science

2018

See the buzzwords!!!
- Streaming
- Calibration/ML
- Distributed Computing
- Heterogeneous
- Statistical Methods

Grand Challenge in Readout and Analysis for Femtoscale Science

Amber Boehnlein, Rolf Ent, Rik Yoshida — November, 2018

Introduction
Micro-electronics and computing technologies have made order-of-magnitude advances in the last decades. Combined with modern statistical methods, it is now possible to analyze scientific data to rapidly expose correlations of data patterns and compare with advanced theoretical models. While many existing nuclear physics and high-energy physics experiments are taking advantage of these developments by upgrading their existing triggered data acquisition to a streaming readout model, these experiments do not have the luxury of an integrated systems from DAQ through analysis. Hence, we aim to remove the separation of data readout and analysis altogether, taking advantage of modern electronics, computing and analysis techniques in order to build the next generation computing model that will be essential for probing femto scale science.

Integrated Whole-Experiment Model
An integrated whole-experiment approach to detector readout and analysis towards scientific output is summarized in the following figure.

Key Elements
An integrated whole-experiment approach to detector readout and analysis towards scientific output will take advantage of multiple existing and emerging technologies. Amongst these are:
- "Streaming readout" where detectors are read out continuously.
- Continuous data quality control and calibration via integration of machine learning technologies.
- Task based high performance local computing.
- Distributed bulk data processing at supercomputer centers.
- Modern statistical methods that can detect differences among groups of data or associations among variables even under very small departures from normality.

Existing and Proposed Efforts
Several of the current LDRD proposals as well as separate on-going efforts naturally fit into the framework of the integrated whole-experiment model of data handling and analysis. They are
- Jefferson Lab EIC science related activities
  - Web-based Pion PDF server
- Jefferson Lab and related part of the Streaming Consortium proposal to the EIC Detector R&D committee including
  - Crate-less streaming prototype
  - TDIS streaming readout prototype
  - EM Calorimeter readout prototype
  - Computing workflow - distributed heterogeneous computing
- LDRD proposals
  - JANA development 2019-LDRD-8
  - Machine Learning MC 2019-LDRD-13
  - Streaming Readout 2019-LDRD-10

Grand Challenge
Develop a proof of concept of quasi-instantaneous high-level nuclear physics analysis based on modern statistics from a self-calibrated matrix of detector raw data synchronized to a reference time, without intermediate data storage requirements, with production systems developed for late stage 12 GeV analysis and the Electron Ion Collider. We propose organizing some of the LDRD proposals and other exploratory work around these themes to achieve proof of concept.

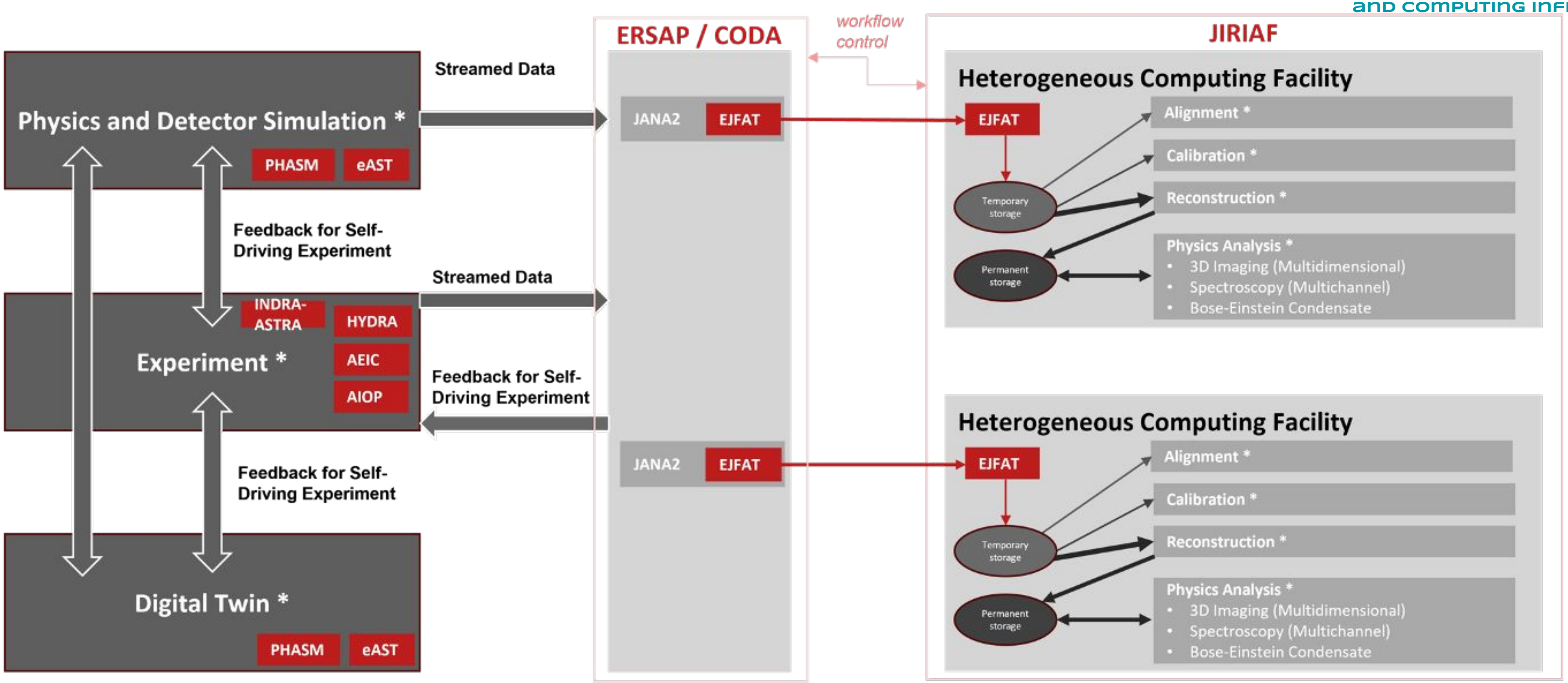
Electron-Ion Collider 4

The Streaming Grand Challenge began in 2018.

Significant progress has been made since then on several fronts that include deployment of SRO-capable fast electronics, firmware development, and software (ERSAP, JANA2, InstaRec,...).

Phase II of the SRO GC is now beginning.

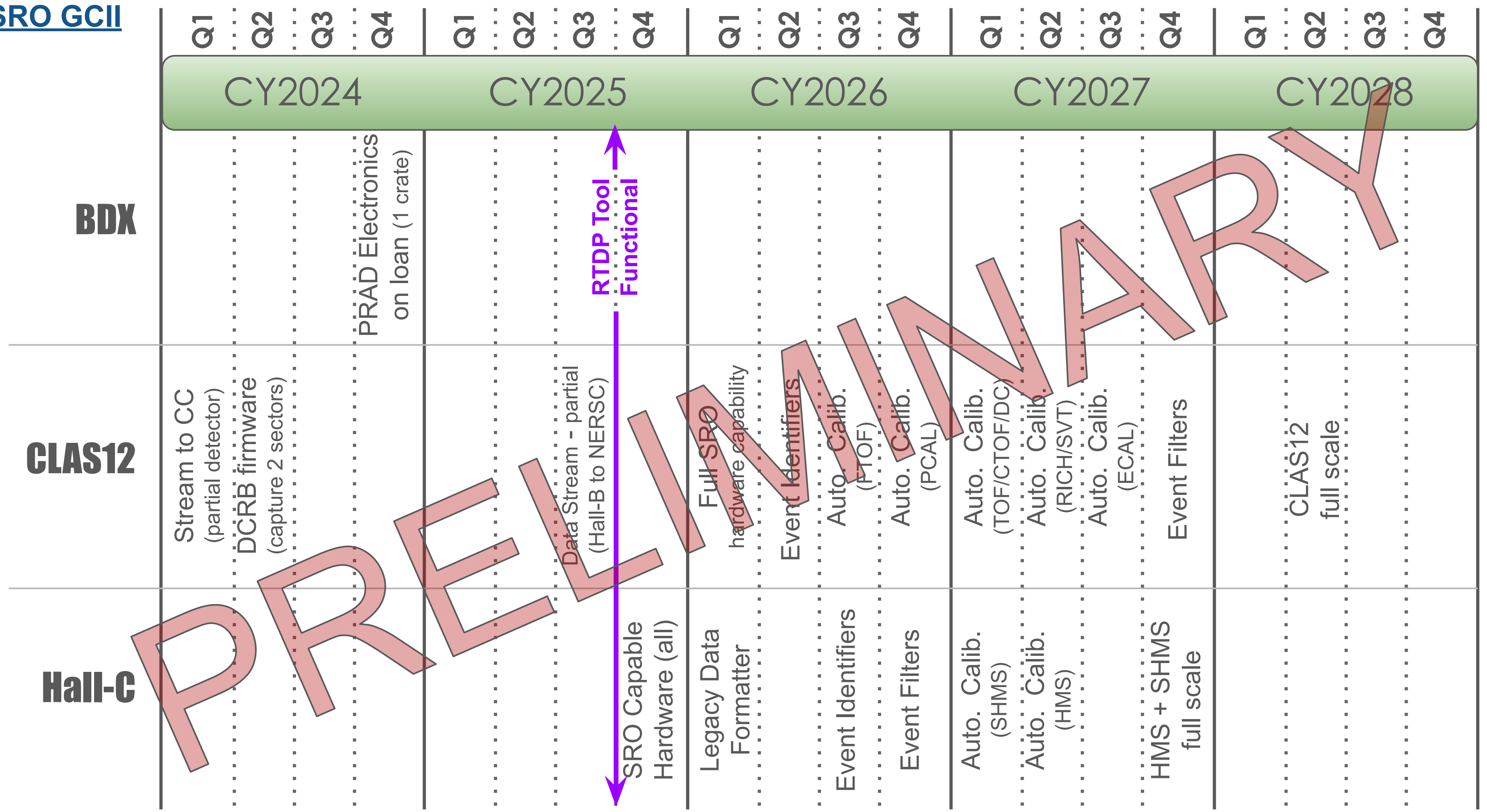
Streaming Grand Challenge Phase II



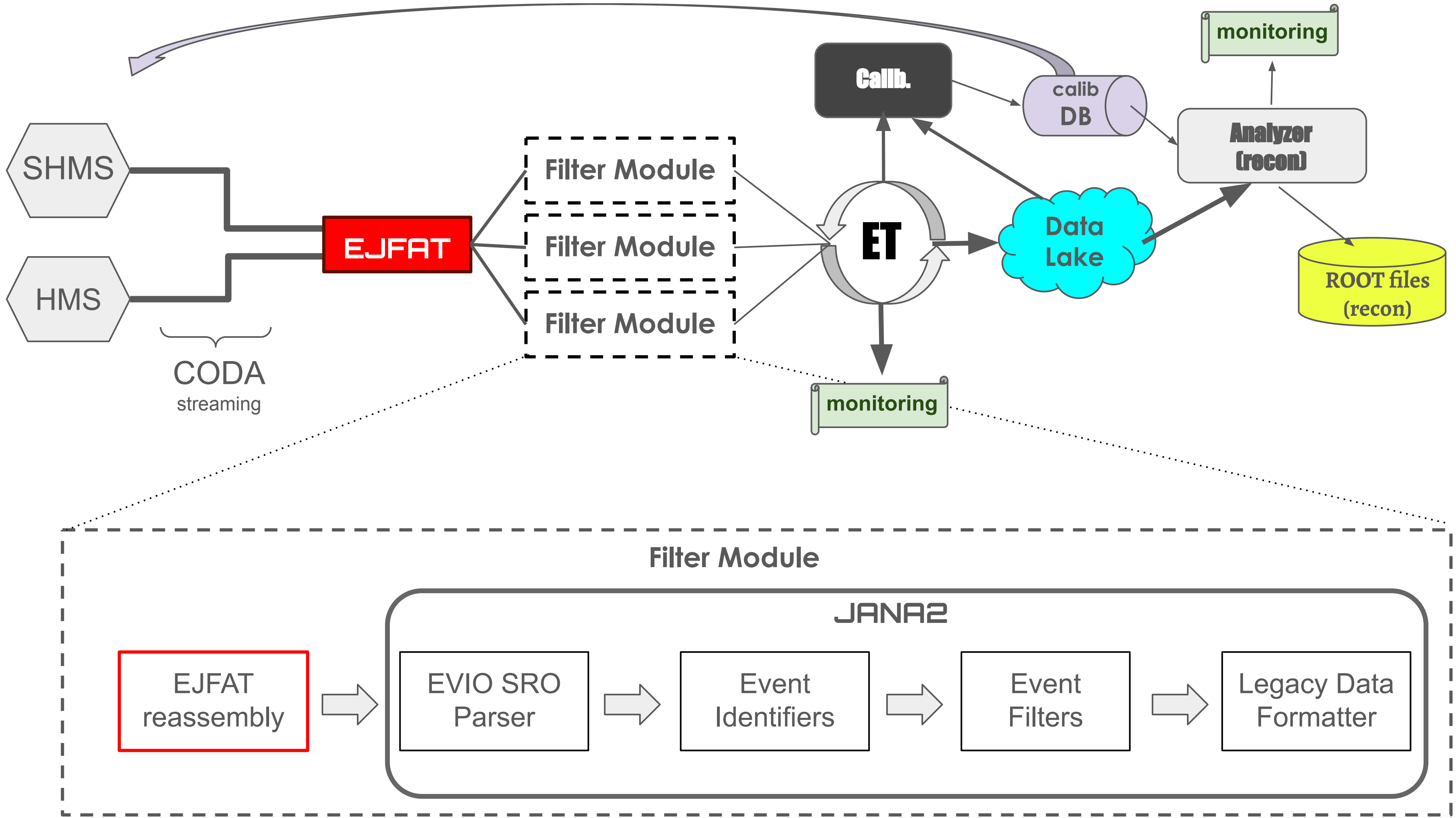
* AI/ML

Diagram from early 2024 by Rolf Ent, Markus Diefenthaler, Brad Sawatsky, and David Lawrence

SRO GCII



Q1 Q2 Q3 Q4
CY2024
CY2025
CY2026
CY2027
CY2028
BDX
CLAS12
Hall-C
PRAD Electronics on loan (1 crate)
RTDP Tool Functional
Stream to CC (partial detector)
DCRB firmware (capture 2 sectors)
Data Stream - partial (Hall-B to NERSC)
Full SRO hardware capability
Event Identifiers
Auto. Calib. (FTOF)
Auto. Calib. (PCAL)
Auto. Calib. (TOF/CTOF/DC)
Auto. Calib. (RICH/SVT)
Auto. Calib. (ECAL)
Event Filters
CLAS12 full scale
SRO Capable Hardware (all)
Legacy Data Formatter
Event Identifiers
Event Filters
Auto. Calib. (SHMS)
Auto. Calib. (HMS)
HMS + SHMS full scale
PRELIMINARY

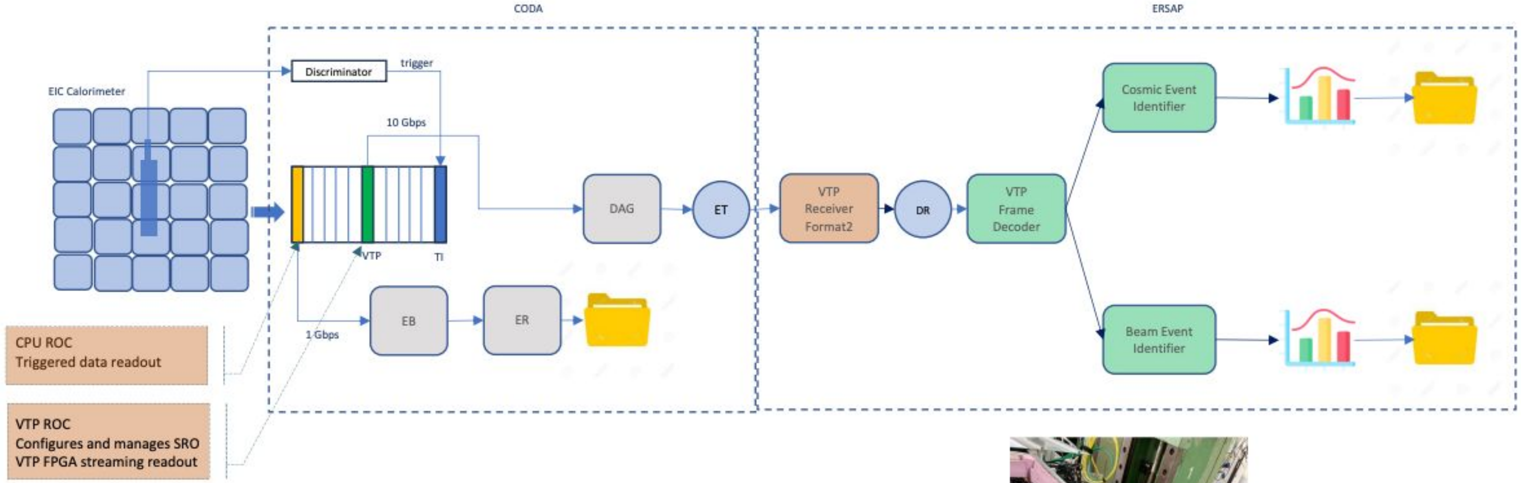


SHMS
HMS
CODA
streaming
EJFAT
Filter Module
Filter Module
Filter Module
ET
Calib.
calib
DB
Data
Lake
Analyzer
(recon)
monitoring
ROOT files
(recon)
monitoring
Filter Module
JANA2
EJFAT
reassembly
EVIO SRO
Parser
Event
Identifiers
Event
Filters
Legacy Data
Formatter

ERSAP - Environment for **R**eal-time **S**treaming, **A**cquisition and **P**rocessing

EPSCI
Experimental Physics Software and Computing Infrastructure

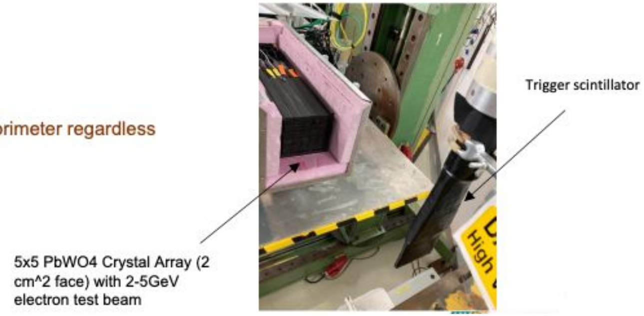
EIC prototype calorimeter SRO pipeline at DESY. CODA & ERSAP



ERSAP design is *event reactive actors*, networked by data pipelines.

- Compositional actors with conditional data routing at runtime.
- Flow-based programming paradigm

Triggered data are waveforms read out over the VME bus.
Stream data are integrated sums and times of all hits over a threshold in the calorimeter regardless of the trigger status.



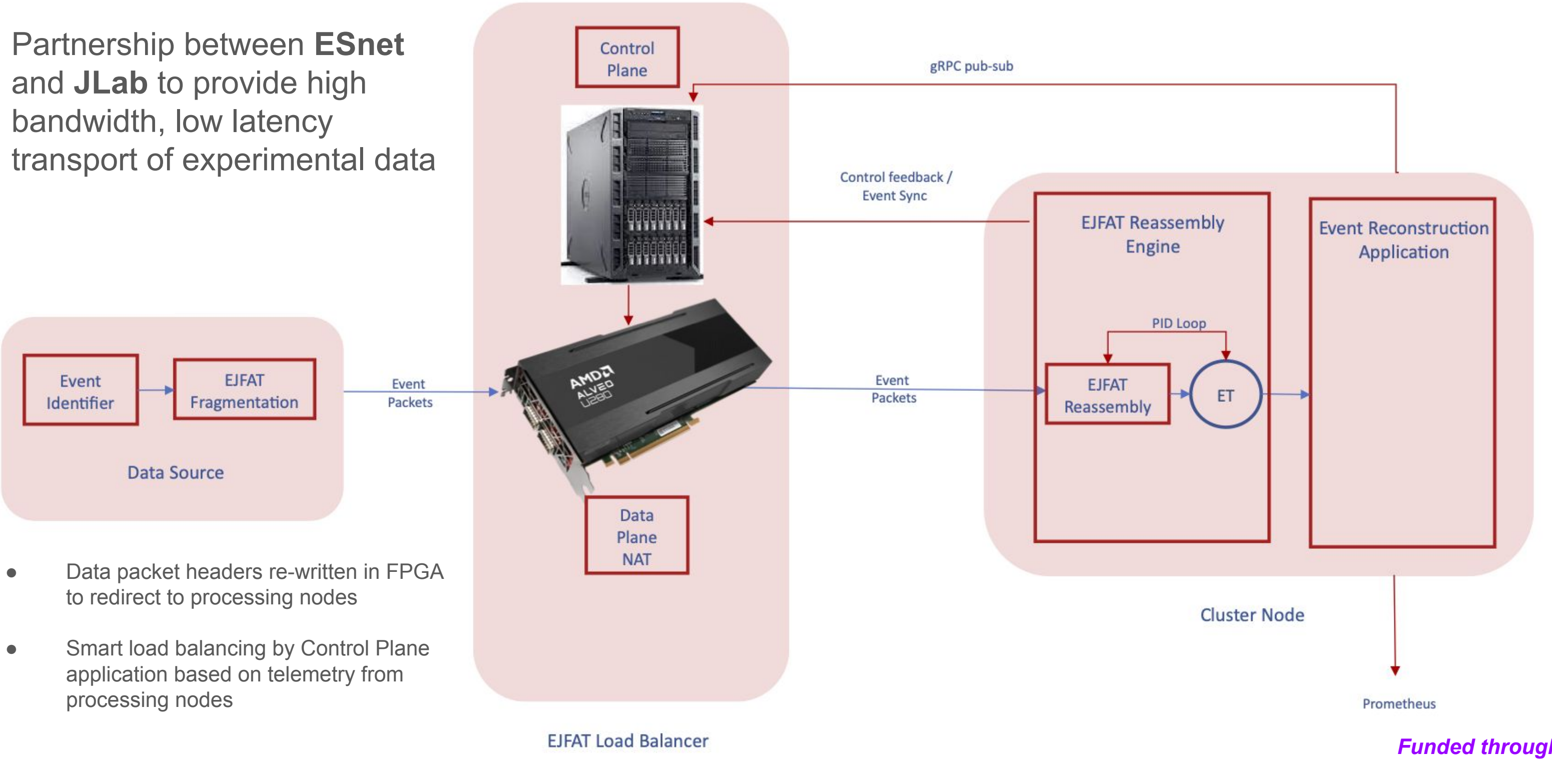
recent beam test at DESY included ERSAP environment for online processing

Funded through JLab SciOPS

ESnet/JLab FPGA Accelerated Transport
(part of IRIAD effort)

EJFAT Block Diagram

Partnership between **ESnet** and **JLab** to provide high bandwidth, low latency transport of experimental data



- Data packet headers re-written in FPGA to redirect to processing nodes
- Smart load balancing by Control Plane application based on telemetry from processing nodes

Funded through DOE ASCR IRIAD project
PI: Graham Heyes
Project Lead: Michael Goodrich

GCII: JIRIAF + EJFAT + NERSC Testing

JIRIAF

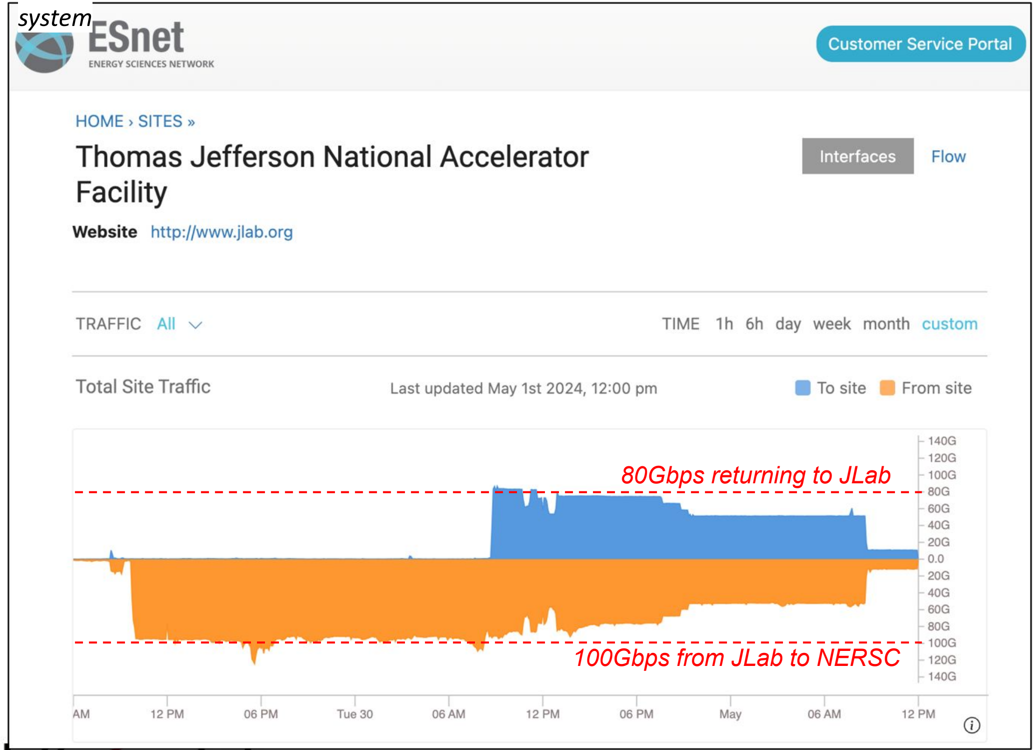
PI: Vardan Gyurjyan
LDRD Project
(funding FY23, FY24)

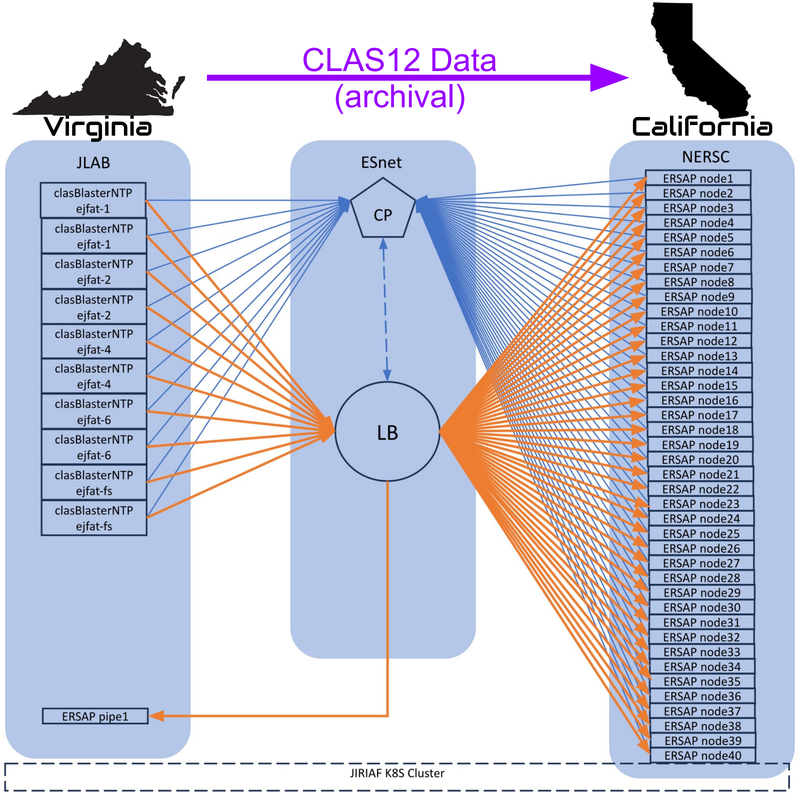
EJFAT
ESnet-Jefferson Lab FPGA Accelerated Transport

PI: Graham Heyes
ASCR Project
Project Lead: Michael Goodrich

EPSCI
Experimental Physics Software and Computing Infrastructure

*Stream exercise sending CLAS12 data from JLab to NERSC using the **JIRIAF** system*

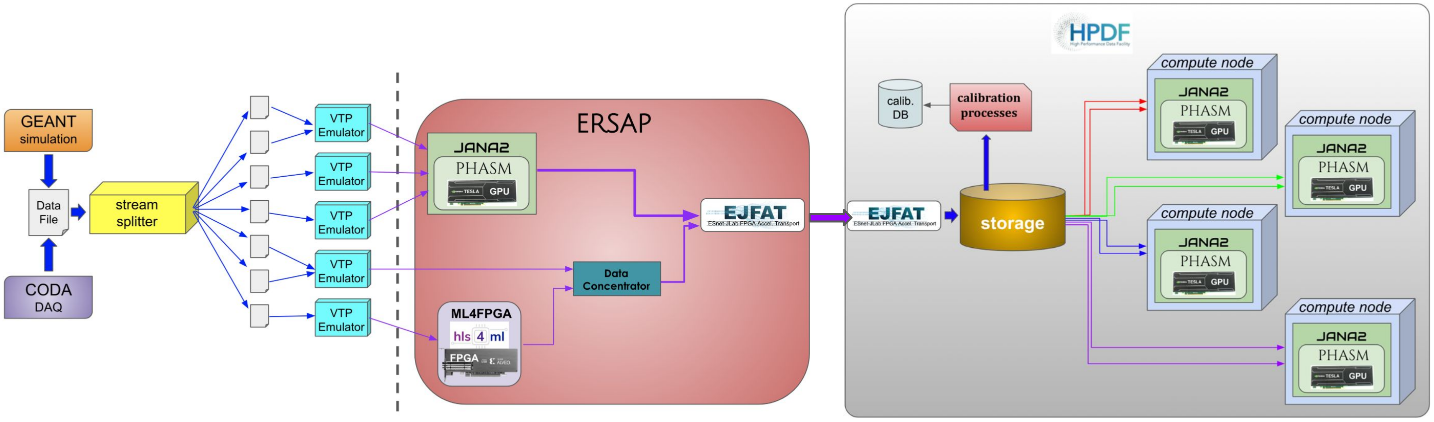




GCII: RTDP - **R**eal-**T**ime **D**evelopment **P**latform

RTDP
SRO Real-Time Development Platform

EPSCI
Experimental Physics Software and Computing Infrastructure



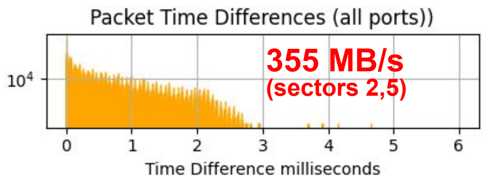
Highly configurable multi-stream source allows realistic streaming simulations

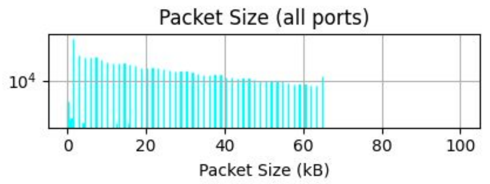
Onsite components will implement first stages of data filtering/reduction

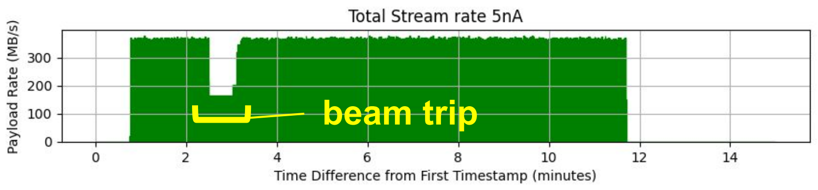
Offsite processing must incorporate built-in calibration latencies and storage. This will also help inform HPDF design

Simulation/emulation of SRO systems with option to replace components with actual hardware or software where available.

May 16th beam test where data streamed from CLAS12 to Data Center and packets captured with hardware timestamps.



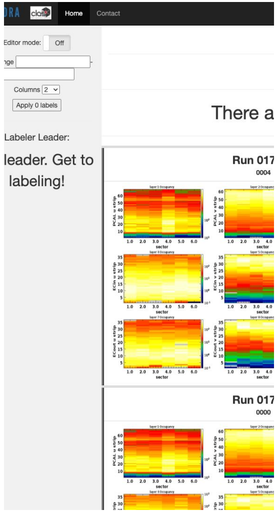




HYDRA Data Quality Monitoring with AI/ML

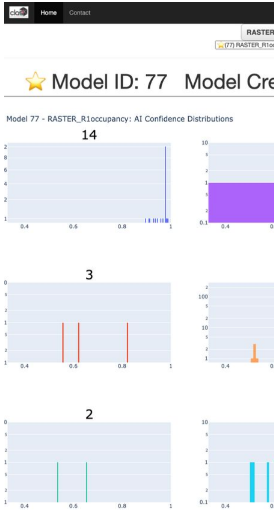
EPSCI
Experimental Physics Software and Computing Infrastructure

HYDRA: Front End



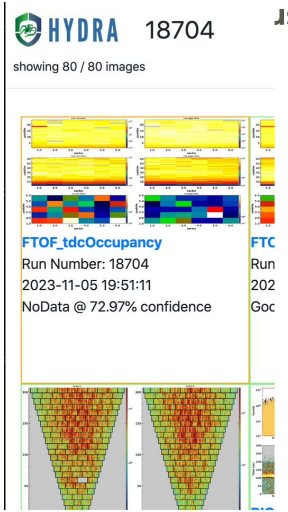
Data Labeler

Efficiently label hundreds (thousands) of images



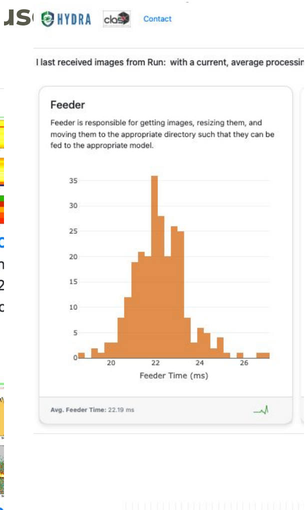
Library

Contains enhanced confusion matrix, thresholds, active model designations



Run

See predictions in real time

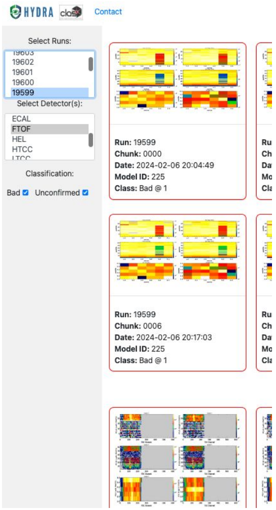


Status

Monitor heartbeats for back end processes and image processing time

Grafana

Dashboard displays all predictions over time



Log

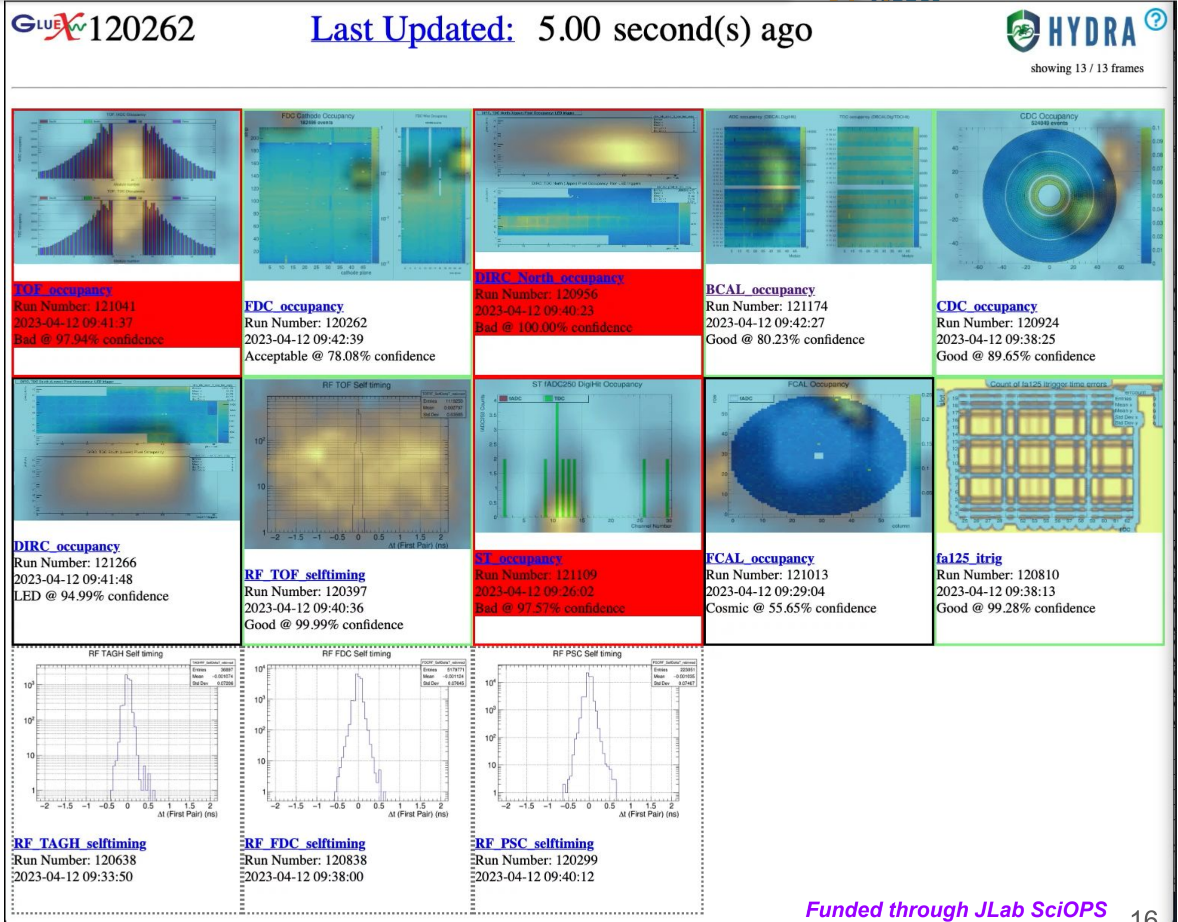
Display concerning plots sorted by detector from previous day

Jefferson Lab
Thomas Jefferson National Accelerator Facility

Funded through JLab SciOPS
Project Lead: Thomas Britton

Data Quality Monitoring with AI/ML

- Models trained on specific plot image
- GradCAM highlighting of problem areas
- Deployed in all 4 experimental halls*



Funded through JLab SciOPS
Project Lead: Thomas Britton

PHASM

What is PHASM?

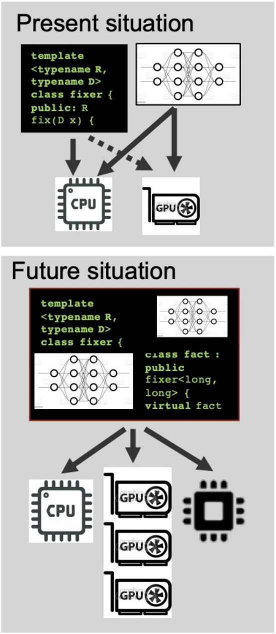
- LDRD project at Jefferson Lab
- 1 year old, 2-3 people
- Proof of concept

Basic Idea

Make it as easy as possible to train a neural net surrogate model to mimic and replace an arbitrary piece of existing numerical code. Systematize and formalize the process from analysis to deployment.

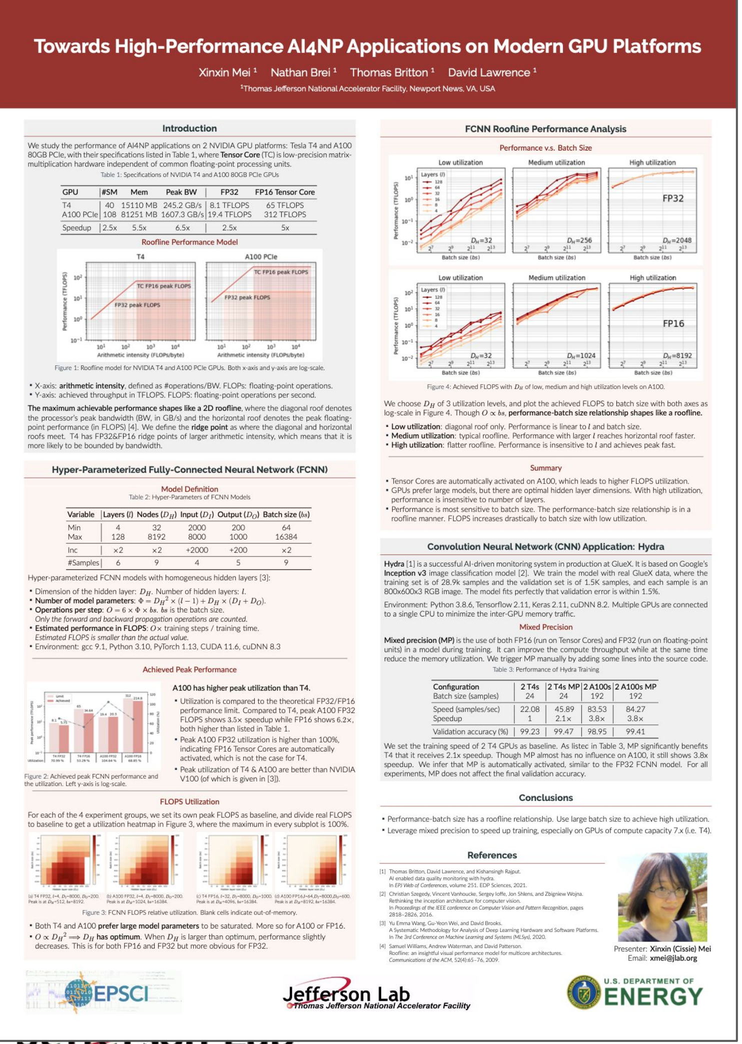
Perspective shift

A neural net surrogate model of an algorithm is a *transformation* of that algorithm. Eventually, classical numerical methods and their data-driven analogues will be understood under a unified theory.



2/14

PI: Nathan Brei
LDRD Project
(funding FY22, FY23)



Towards High-Performance AI4NP Applications on Modern GPU Platforms

Xinxin Mei [1] Nathan Brei [1] Thomas Britton [1] David Lawrence [1]

[1]Thomas Jefferson National Accelerator Facility, Newport News, VA, USA

Introduction

We study the performance of AI4NP applications on 2 NVIDIA GPU platforms: Tesla T4 and A100 80GB PCIe, with their specifications listed in Table 1, where **Tensor Core** (TC) is low-precision matrix-multiplication hardware independent of common floating-point processing units.

Table 1: Specifications of NVIDIA T4 and A100 80GB PCIe GPUs

GPU	#SM	Mem	Peak BW	FP32	FP16 Tensor Core
T4	40	15110 MB	245.2 GB/s	8.1 TFLOPS	65 TFLOPS
A100 PCIe	108	81251 MB	1607.3 GB/s	19.4 TFLOPS	312 TFLOPS
Speedup	2.5x	5.5x	6.5x	2.5x	5x

Roofline Performance Model

Figure 1: Roofline model for NVIDIA T4 and A100 PCIe GPUs. Both x-axis and y-axis are log-scale.

- X-axis: **arithmetic intensity**, defined as #operations/BW. FLOPs: floating-point operations.
- Y-axis: achieved throughput in TFLOPS. FLOPS: floating-point operations per second.

The maximum achievable performance shapes like a 2D roofline, where the diagonal roof denotes the processor's peak bandwidth (BW, in GB/s) and the horizontal roof denotes the peak floating-point performance (in FLOPS) [4]. We define the **ridge point** as where the diagonal and horizontal roofs meet. T4 has FP32&FP16 ridge points of larger arithmetic intensity, which means that it is more likely to be bounded by bandwidth.

Hyper-Parameterized Fully-Connected Neural Network (FCNN)

Model Definition

Table 2: Hyper-Parameters of FCNN Models

Variable	Layers (l)	Nodes (D_H)	Input (D_I)	Output (D_O)	Batch size (bs)
Min	4	32	2000	200	64
Max	128	8192	8000	1000	16384
Inc	×2	×2	+2000	+200	×2
#Samples	6	9	4	5	9

Hyper-parameterized FCNN models with homogeneous hidden layers [3]:

- Dimension of the hidden layer: D_H. Number of hidden layers: l.
- **Number of model parameters**: $\Phi = {D_H}^2 \times (l-1) + D_H \times (D_I + D_O)$.
- **Operations per step**: $O = 6 \times \Phi \times bs$. bs is the batch size.
  Only the forward and backward propagation operations are counted.
- **Estimated performance in FLOPS**: $O\times$ training steps / training time.
  Estimated FLOPS is smaller than the actual value.
- Environment: gcc 9.1, Python 3.10, PyTorch 1.13, CUDA 11.6, cuDNN 8.3

Achieved Peak Performance

Figure 2: Achieved peak FCNN performance and the utilization. Left y-axis is log-scale.

A100 has higher peak utilization than T4.

- Utilization is compared to the theoretical FP32/FP16 performance limit. Compared to T4, peak A100 FP32 FLOPS shows 3.5× speedup while FP16 shows 6.2×, both higher than listed in Table 1.
- Peak A100 FP32 utilization is higher than 100%, indicating FP16 Tensor Cores are automatically activated, which is not the case for T4.
- Peak utilization of T4 & A100 are better than NVIDIA V100 (of which is given in [3]).

FLOPS Utilization

For each of the 4 experiment groups, we set its own peak FLOPS as baseline, and divide real FLOPS to baseline to get a utilization heatmap in Figure 3, where the maximum in every subplot is 100%.

(a) T4 FP32, l=4, D_I=8000, D_O=200. Peak is at D_H=512, bs=8192.
(b) A100 FP32, l=4, D_I=8000, D_O=200. Peak is at D_H=1024, bs=16384.
(c) T4 FP16, l=32, D_I=8000, D_O=1000. Peak is at D_H=4096, bs=16384.
(d) A100 FP16 l=64, D_I=8000, D_O=600. Peak is at D_H=8192, bs=16384.

Figure 3: FCNN FLOPS relative utilization. Blank cells indicate out-of-memory.

- Both T4 and A100 **prefer large model parameters** to be saturated. More so for A100 or FP16.
- $O \propto {D_H}^2 \Longrightarrow D_H$ **has optimum**. When D_H is larger than optimum, performance slightly decreases. This is for both FP16 and FP32 but more obvious for FP32.

FCNN Roofline Performance Analysis

Performance v.s. Batch Size

Figure 4: Achieved FLOPS with D_H of low, medium and high utilization levels on A100.

We choose D_H of 3 utilization levels, and plot the achieved FLOPS to batch size with both axes as log-scale in Figure 4. Though $O \propto bs$, **performance-batch size relationship shapes like a roofline.**

- **Low utilization**: diagonal roof only. Performance is linear to l and batch size.
- **Medium utilization**: typical roofline. Performance with larger l reaches horizontal roof faster.
- **High utilization**: flatter roofline. Performance is insensitive to l and achieves peak fast.

Summary

- Tensor Cores are automatically activated on A100, which leads to higher FLOPS utilization.
- GPUs prefer large models, but there are optimal hidden layer dimensions. With high utilization, performance is insensitive to number of layers.
- Performance is most sensitive to batch size. The performance-batch size relationship is in a roofline manner. FLOPS increases drastically to batch size with low utilization.

Convolution Neural Network (CNN) Application: Hydra

Hydra [1] is a successful AI-driven monitoring system in production at GlueX. It is based on Google's **Inception v3** image classification model [2]. We train the model with real GlueX data, where the training set is of 28.9k samples and the validation set is of 1.5K samples, and each sample is an 800x600x3 RGB image. The model fits perfectly that validation error is within 1.5%.

Environment: Python 3.8.6, Tensorflow 2.11, Keras 2.11, cuDNN 8.2. Multiple GPUs are connected to a single CPU to minimize the inter-GPU memory traffic.

Mixed Precision

Mixed precision (MP) is the use of both FP16 (run on Tensor Cores) and FP32 (run on floating-point units) in a model during training. It can improve the compute throughput while at the same time reduce the memory utilization. We trigger MP manually by adding some lines into the source code.

Table 3: Performance of Hydra Training

Configuration	2 T4s	2 T4s MP	2 A100s	2 A100s MP
Batch size (samples)	24	24	192	192
Speed (samples/sec)	22.08	45.89	83.53	84.27
Speedup	1	2.1×	3.8×	3.8×
Validation accuracy (%)	99.23	99.47	98.95	99.41

We set the training speed of 2 T4 GPUs as baseline. As listec in Table 3, MP significantly benefits T4 that it receives 2.1x speedup. Though MP almost has no influence on A100, it still shows 3.8x speedup. We infer that MP is automatically activated, similar to the FP32 FCNN model. For all experiments, MP does not affect the final validation accuracy.

Conclusions

- Performance-batch size has a roofline relationship. Use large batch size to achieve high utilization.
- Leverage mixed precision to speed up training, especially on GPUs of compute capacity 7.x (i.e. T4).

References

[1] Thomas Britton, David Lawrence, and Kishansingh Rajput. AI enabled data quality monitoring with hydra. In *EPJ Web of Conferences*, volume 251. EDP Sciences, 2021.

[2] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[3] Yu Emma Wang, Gu-Yeon Wei, and David Brooks. A Systematic Methodology for Analysis of Deep Learning Hardware and Software Platforms. In *The 3rd Conference on Machine Learning and Systems (MLSys)*, 2020.

[4] Samuel Williams, Andrew Waterman, and David Patterson. Roofline: an insightful visual performance model for multicore architectures. *Communications of the ACM*, 52(4):65–76, 2009.

Presenter: **Xinxin (Cissie) Mei**
Email: **xmei@jlab.org**

GPU Benchmarking for AI4NP applications

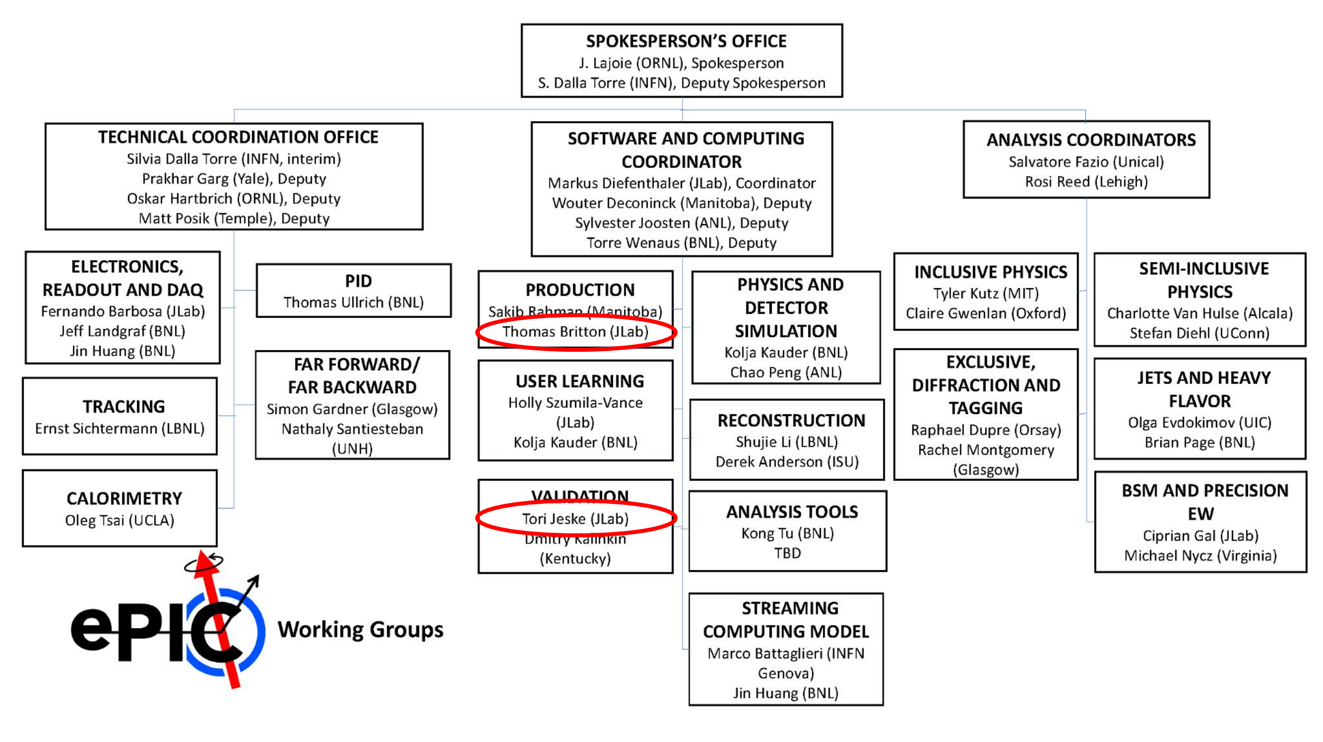
- Roofline model analysis of two types of Nvidia GPUs with 2 AI applications
- Generic MLP model
- Google Inception v3 (used by Hydra)
- FP16, FP32, and Mixed Precision

https://github.com/cissieAB/pytorch-paradnn/blob/master/docs/CHEP_GPU4ML4NP_05022023.pdf

PI: Nathan Brei
Task Lead: Cissie Mei
LDRD Project
(funding FY22, FY23)

Electron Ion Collider

EPSCI
Experimental Physics Software and Computing Infrastructure



HSF
HEP Software Foundation

Conveners (Frameworks)

The group conveners are:

- Nathan Brei, EIC and JLab
- Benedikt Hegner, Key4hep and CERN
- Kyle Knoepfel, Neutrino Platform and Fermilab
- Vakho Tsulaia, ATLAS and LBNL

Code Maintenance Responsibilities

EPSCI
Experimental Physics Software and Computing Infrastructure

CEBAF Large Acceptance Spectrometer

CLARA Reconstruction Framework

JANA Reconstruction Framework

CODA

Data Acquisition System

OSG

MCWrapper Simulation Workflow Manager

Funded through JLab SciOPS

Publications

https://wiki.jlab.org/epsciwiki/index.php/EPSCI_publications_page

EPSCI
hysics software
s infrastructure

2020 [edit]

Title/Info
Automated and Distributed
Thomas Britton
EPJ Web Conf., 245 (2020) 01022
DOI: https://doi.org/10.1051/epjconf/2
JANA2 Framework for Eve
David Lawrence, Amber Boehnlein an
EPJ Web Conf., 245 (2020) 01022
DOI: https://doi.org/10.1051/epjconf/2
ML Track Fitting in Nuclear
Thomas Britton, David Lawrence and
EPJ Web Conf., 245 (2020) 06015
DOI: https://doi.org/10.1051/epjconf/2
Offsite Data Processing for
David Lawrence
EPJ Web Conf., 245 (2020) 07037
DOI: https://doi.org/10.1051/epjconf/2
Heterogeneous data-proce
Vardan Gyurjyan and Sebastian Mancilla
EPJ Web Conf., 245 (2020) 07037
DOI: https://doi.org/10.1051/epjconf/202024505020

2021 [edit]

Title/Info
Uncertainty aware anomaly
Willem Blokland, Pradeep Ramuhalli,
Submitted to *Phy AB*
arXiv: https://arxiv.org/abs/2110.1200
AI Enabled Data Quality M
Thomas Britton, David Lawrence, Kish
DOI: https://doi.org/10.1051/epjconf/2
arXiv: https://arxiv.org/abs/2105.0794
HOSS!
David Lawrence
DOI: https://doi.org/10.1051/epjconf/2
arXiv: https://arxiv.org/abs/2104.1465
Streaming Readout of the
Fabrizio Ameli, Marco Battaglieri, Mar
, Vardan Gyurjyan, David Lawrence, F
DOI: https://doi.org/10.1051/epjconf/2
arXiv: https://arxiv.org/abs/2105.0794

2022 [edit]

Title/Info
EJ-FAT Joint ESnet JLab FPGA A
Stacey Sheldon, Yatish Kumar, Michael Goodric
Submitted to INDIS
arXiv: https://arxiv.org/abs/2303.16351
Scientific Computing Plan for the
J. C. Bernauer, et al.
Will be submitted to NIM
arXiv: https://doi.org/10.48550/arXiv.2205.0860
NIM: https://www.sciencedirect.com/science/arti
Accelerator and detector control f
T. Britton and B. Nachman
Submitted to IOPscience Journal of Instrumenta
DOI: https://doi.org/10.1088/1748-0221/17/02/C
Using AI to predict calibrations co
Torri Jeske, Diana McSpadden, Nikhil Kalra, Th
Submitted to IOPscience Journal of Physics: Co
AI For Experimental Controls at Je
Torri Jeske, Diana McSpadden, Nikhil Kalra, Th
DOI: https://doi.org/10.1088/1748-0221/17/03/C
arXiv: https://arxiv.org/abs/2203.05999
Machine Learning on FPGA For E
Sergey Furletov, Fernando Barbosa, Lee Belfore
Submitted to JINST
Streaming readout for next genera
Fabrizio Ameli, Marco Battaglieri, Vladimir V. Berdnikov, Mariangela Bondi, Sergey Boyarinov, Nath
, Tommaso Chiarusi, Raffaella De Vita, Cristiano Fanelli, Vardan Gyurjyan, David Lawrence, Patrick
, Alessandro Pilloni, Ben Raydo, Carl Timmer, Maurizio Ungaro, Simone Vallarino
Submitted to EPJ+
DOI: https://link.springer.com/article/10.1140/epjp/s13360-022-03146-z

2023 [edit]

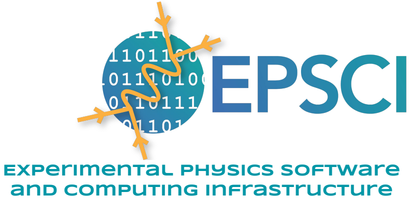
Title/Info
Control Simulation for an ESnet-JLa
D. Howard; M. Goodrich; C. Timmer; V. Gyurjyan;
Published at: [(pending)]
DOI: [https://jeffersonlab.sharepoint.com/:b:/r/sites
Development of ML FPGA Filter for
F. Barbosa, L. Belfore, N. Branson, C. Dickover, C
Published at: IEEE Xplore
DOI: https://doi.org/10.1109/TNS.2023.3259436
EJFAT: Towards Intelligent Compute
M. Goodrich; C. Timmer; V. Gyurjyan; D. Lawrenc
Submitted to: ACAT
DOI: https://indico.cern.ch/event/1106990/contrib
The Present and Future of QCD
(report from 2022 Hot & Cold QCD To
P. Achenbach et al.
Submitted to: arXiv.org
DOI: https://doi.org/10.48550/arXiv.2303.02579
ESnet / JLab FPGA Accelerated Tra
M. Goodrich; C. Timmer; V. Gyurjyan; D. Lawrenc
Submitted to: IEEE Xplore
DOI: https://doi.org/10.1109/TNS.2023.3243871
ERSAP: Towards Better NP Data-S
V. Gyurjyan, D. Abbott, N. Brei, M. Goodrich, G. H
Published at: IEEE Xplore
DOI: https://doi.org/10.1109/TNS.2023.3242548

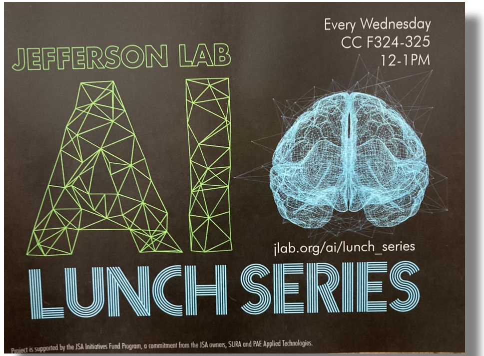
2024 [edit]

Publications from 2024

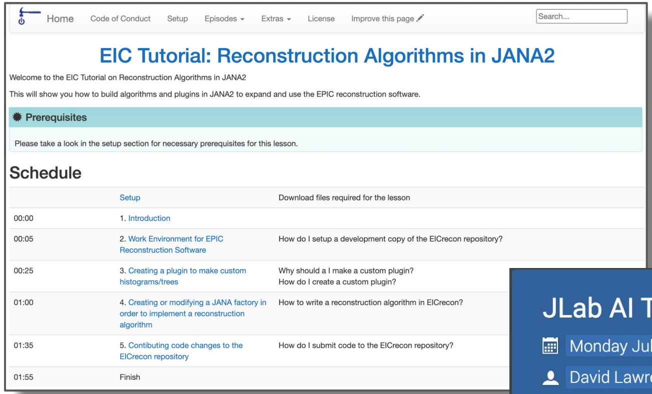
Title/Info	BibTex
Towards High-Performance AI4NP Applications on Modern GPU Platforms Xinxin Mei, Nathan Brei and David Lawrence EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429511023	bibtex
JIRIAF: JLAB Integrated Research Infrastructure Across Facilities Gyurjyan Vardan, Larrieu Christopher, Heyes Graham, and Lawrence David EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429504027	bibtex
EIC Software Overview Lawrence David EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429503011	bibtex
Streaming Readout and Data-Stream Processing With ERSAP Gyurjyan Vardan, Abbott David, Goodrich Michael, Heyes Graham, Jastrzembski Ed, Lawrence David, Raydo Benjamin and Timmer Carl EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429502025	bibtex
Hydra: Computer Vision for Online Data Quality Monitoring Torri Jeske, Thomas Britton, David Lawrence and Kishansingh Rajput EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429502008	bibtex
AI Driven Experiment Calibration and Control Thomas Britton, Cullan Bedwell3, Abhijeet Chawhan3, Julie Crowe3, Naomi Jarvis2, Torri Jeske1, Nikhil Kalra1, David Lawrence, and Diana McSpadden EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429502003	bibtex
Control Simulation for an ESnet-JLab FPGA Accelerated Transport Load Balancer Derek Howard, Michael Goodrich, Carl Timmer, Yatish Kumar, Graham Heyes, David Lawrence, Stacey Sheldon, and Vardan Gyurjyan EPJ Web of Conferences https://doi.org/10.1051/epjconf/202429510002	bibtex
Hydra: Computer Vision for Data Quality Monitoring Thomas Britton, Torri Jeske, David Lawrence, Kishansingh Rajput	

Education and Outreach





EIC Tutorials


Home | Code of Conduct | Setup | Episodes | Extras | License | Improve this page | Search...

EIC Tutorial: Reconstruction Algorithms in JANA2

Welcome to the EIC Tutorial on Reconstruction Algorithms in JANA2

This will show you how to build algorithms and plugins in JANA2 to expand and use the EPIC reconstruction software.

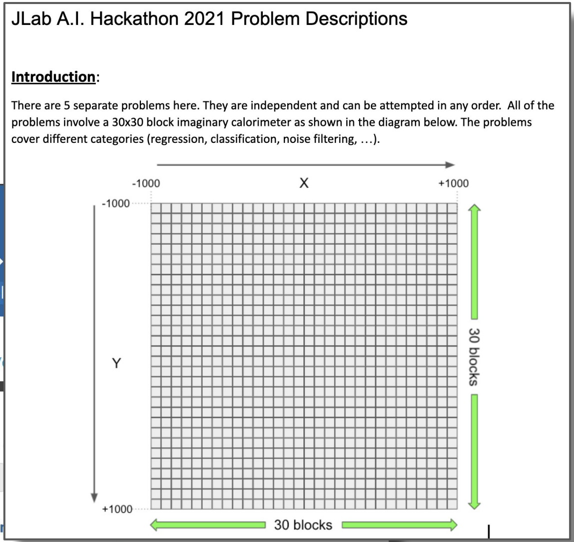
Prerequisites

Please take a look in the setup section for necessary prerequisites for this lesson.

Schedule

	Setup	Download files required for the lesson
00:00	1. Introduction	
00:05	2. Work Environment for EPIC Reconstruction Software	How do I setup a development copy of the EICrecon repository?
00:25	3. Creating a plugin to make custom histograms/trees	Why should a I make a custom plugin? How do I create a custom plugin?
01:00	4. Creating or modifying a JANA factory in order to implement a reconstruction algorithm	How to write a reconstruction algorithm in EICrecon?
01:35	5. Contibuting code changes to the EICrecon repository	How do I submit code to the EICrecon repository?
01:55	Finish	

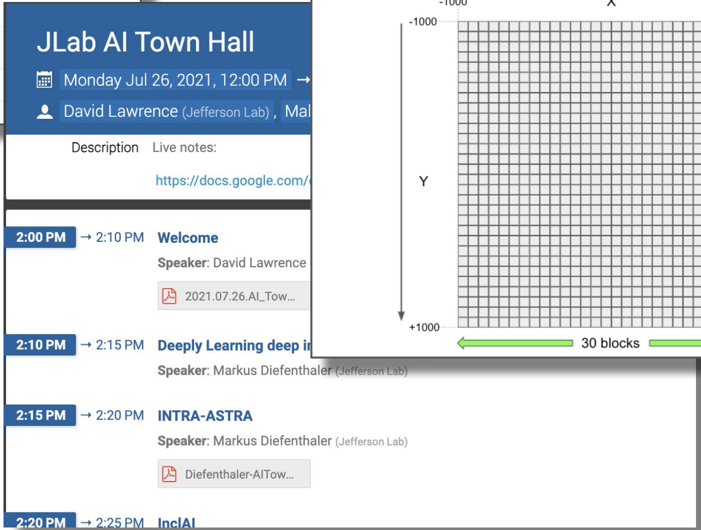
AI/ML Hackathons


JLab A.I. Hackathon 2021 Problem Descriptions

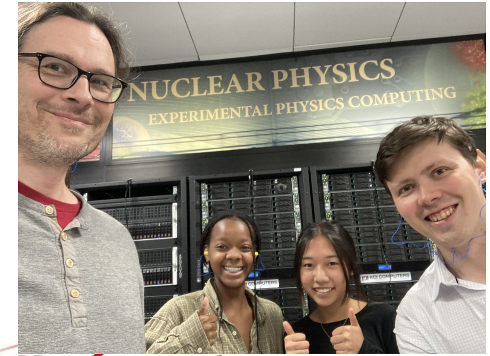
Introduction:

There are 5 separate problems here. They are independent and can be attempted in any order. All of the problems involve a 30x30 block imaginary calorimeter as shown in the diagram below. The problems cover different categories (regression, classification, noise filtering, …).

Town Halls



Summer Students



Summary

- EPSCI is a multidisciplinary team with Software and Computing expertise
- Our goal is to help develop and apply new technologies to the ENP program
- Experimental Hall involvement:
  - Involved in all halls via CODA/EVIO, Hydra
  - Strong involvement in Hall-B and Hall-D offline (working to get more involved in Halls A&C)
  - Strong involvement in ePIC as WG conveners and through JANA
- Major projects:
  - Support ENP systems we helped develop (CLARA*, JANA, CODA, MCWrapper, …)
  - Streaming Data Acquisition (ERSAP, RTDP, SRO Grand Challenge)
  - Offsite data processing (EJFAT, JIRIAF)
  - AI/ML Data Quality Monitoring (Hydra)
  - AI/ML Experiment Controls (AIEC[2021-2023], AIOP[2024-2026])
  - R&D (PHASM*)

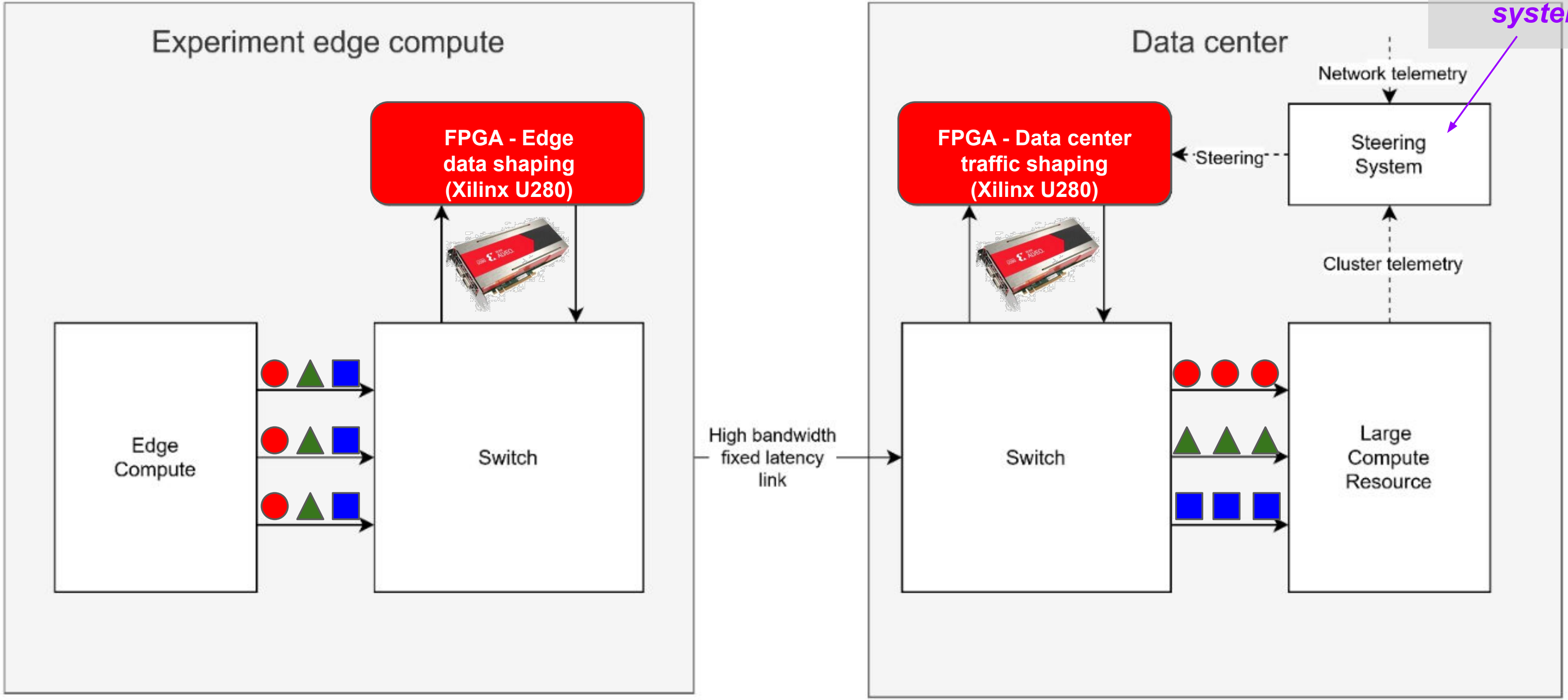
EXPERIMENTAL PHYSICS SOFTWARE AND COMPUTING INFRASTRUCTURE

Backups

EJFAT Concept: Edge to Data Center Traffic Shaping / Steering

ASCR Funded Project

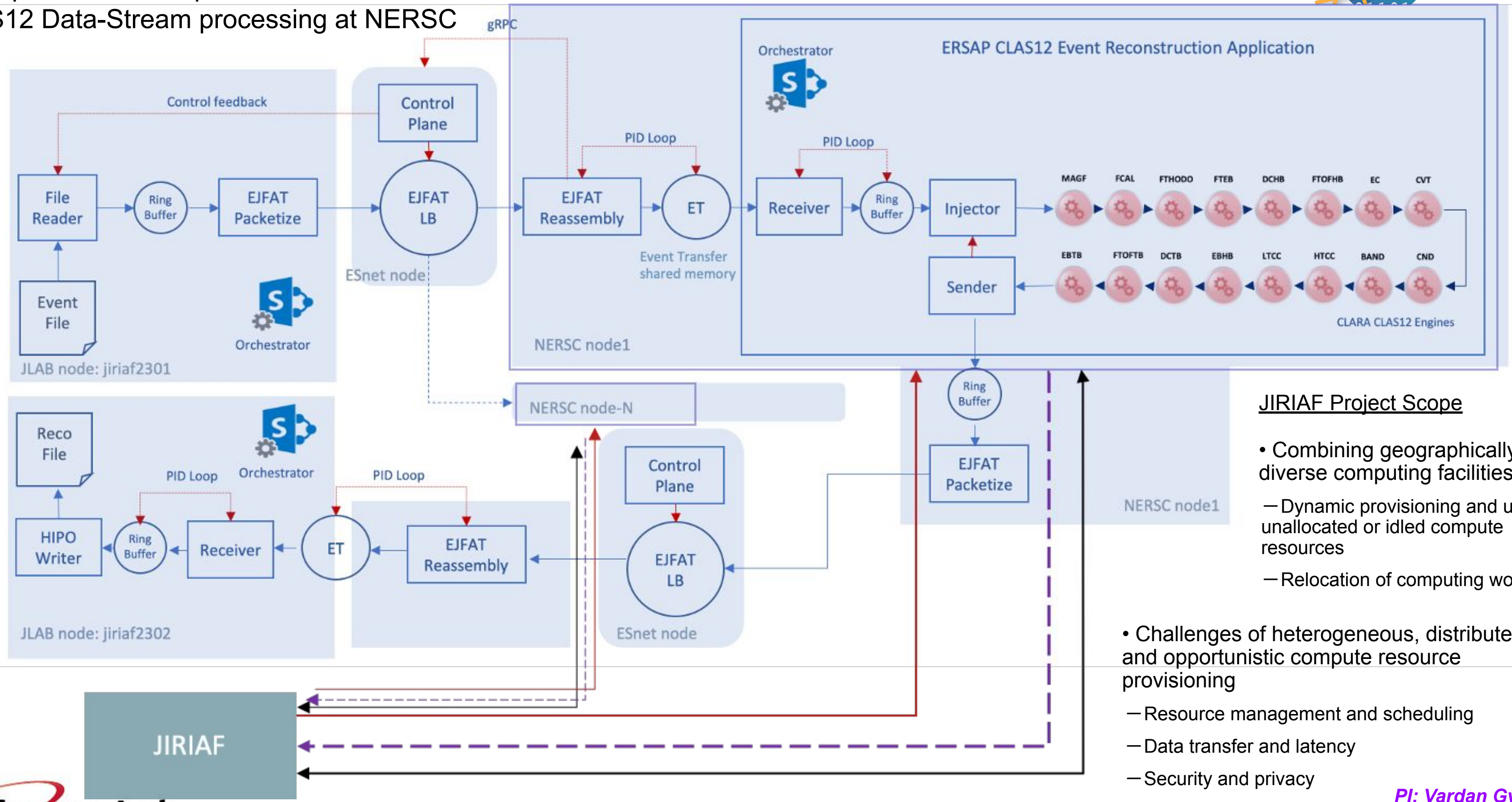
potential AI system



EJFAT = ESnet + JLab FPGA Accelerated Transport

JLAB Integrated Research Infrastructure Across Facilities

Concept Validation Experiment:
CLAS12 Data-Stream processing at NERSC



<u>JIRIAF Project Scope</u>

- Combining geographically diverse computing facilities
  - Dynamic provisioning and use of unallocated or idled compute resources
  - Relocation of computing workflows
- Challenges of heterogeneous, distributed, and opportunistic compute resource provisioning
  - Resource management and scheduling
  - Data transfer and latency
  - Security and privacy

PI: Vardan Gyurjyan
LDRD Project
(funding FY23, FY24)