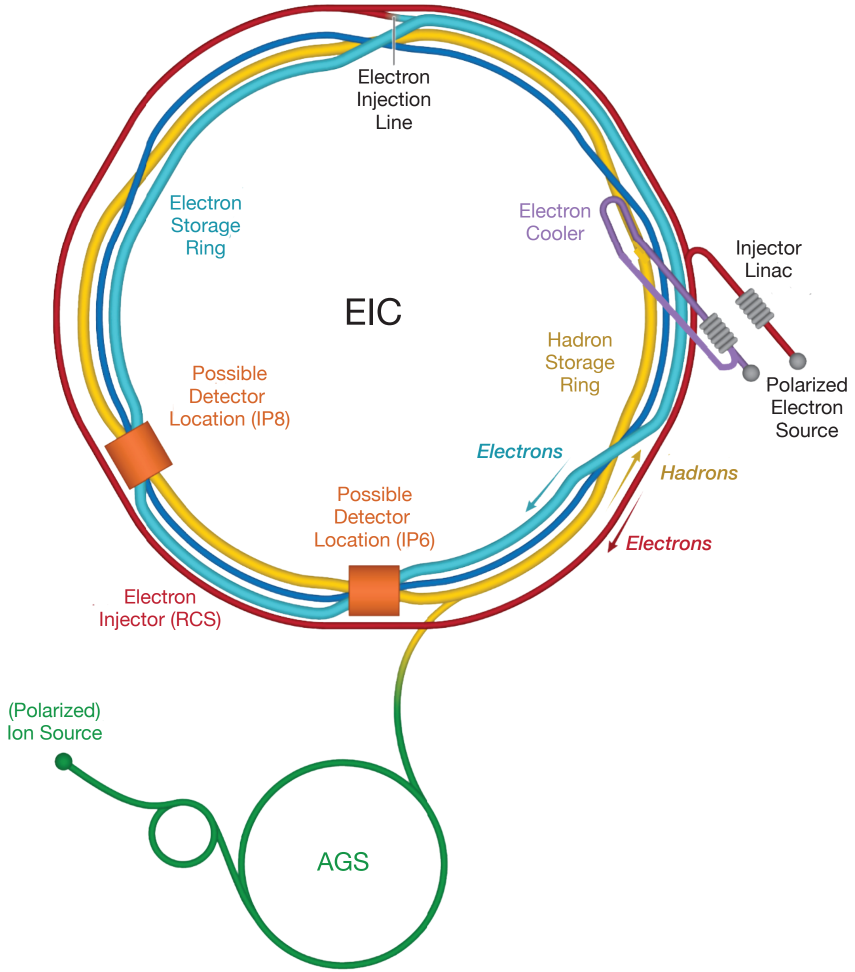
Electron-Ion Collider:
Software & Computing

Markus Diefenthaler
EIC^2 at Jefferson Lab

The Electron-Ion Collider (EIC)



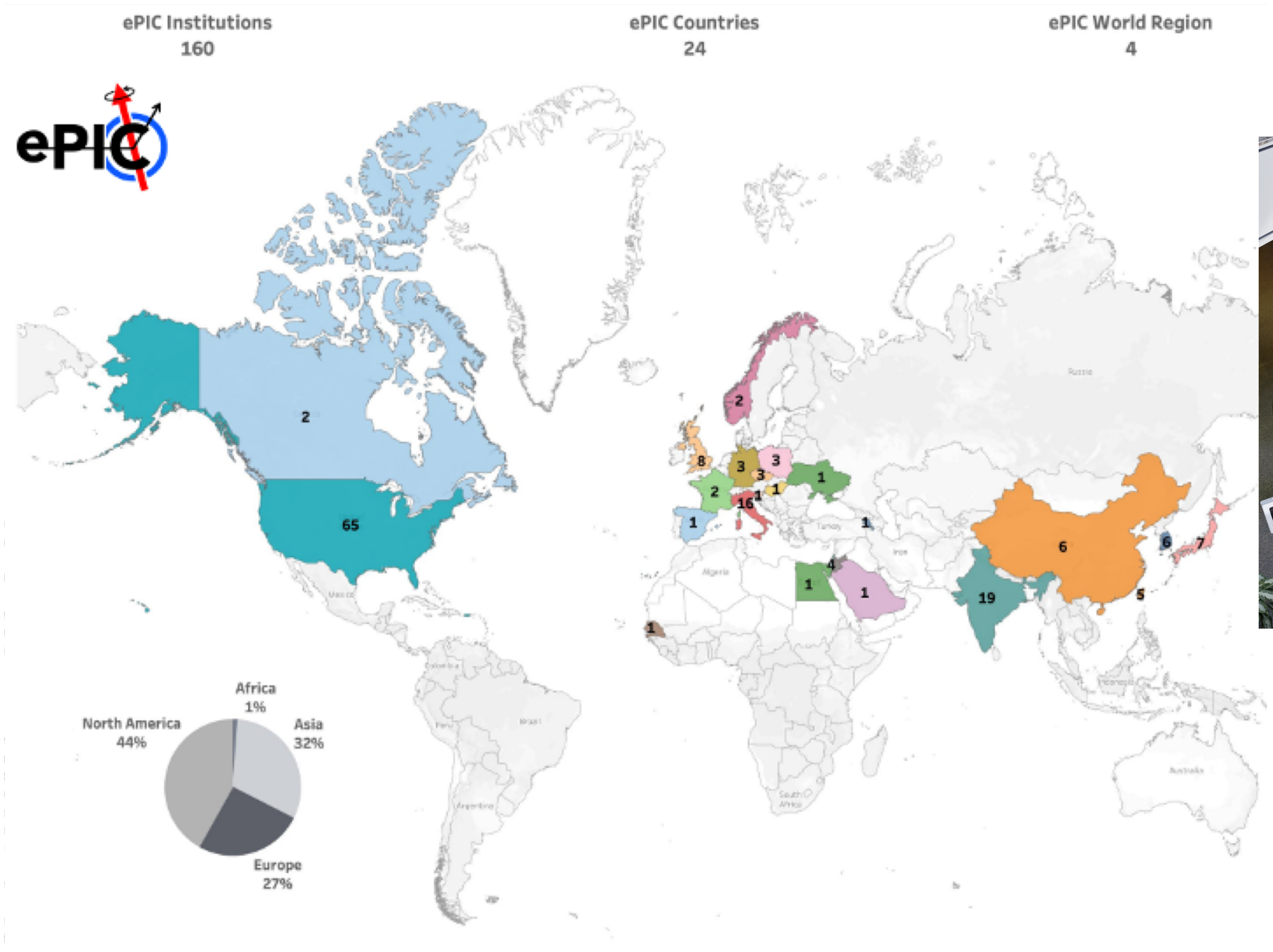
Frontier accelerator facility in the U.S.

- **World's first collider of**:
  - Polarized electrons and polarized protons,
  - Polarized electrons and light ions (d, ^{3}He),
  - Electrons and heavy ions (up to Uranium).

- The EIC will enable us to embark on a **precision study of the nucleon and the nucleus at the scale of sea quarks and gluons**, over all of the kinematic range that is relevant.
- The **EIC Yellow Report** (Nucl.Phys.A 1026 (2022) 122447) describes the physics case, the resulting detector requirements, and the evolving detector concepts for the experimental program at the EIC.

- BNL and Jefferson Lab will be host laboratories for the EIC Experimental Program. Leadership roles in the EIC project are shared.
- EIC operations will start in about a decade.

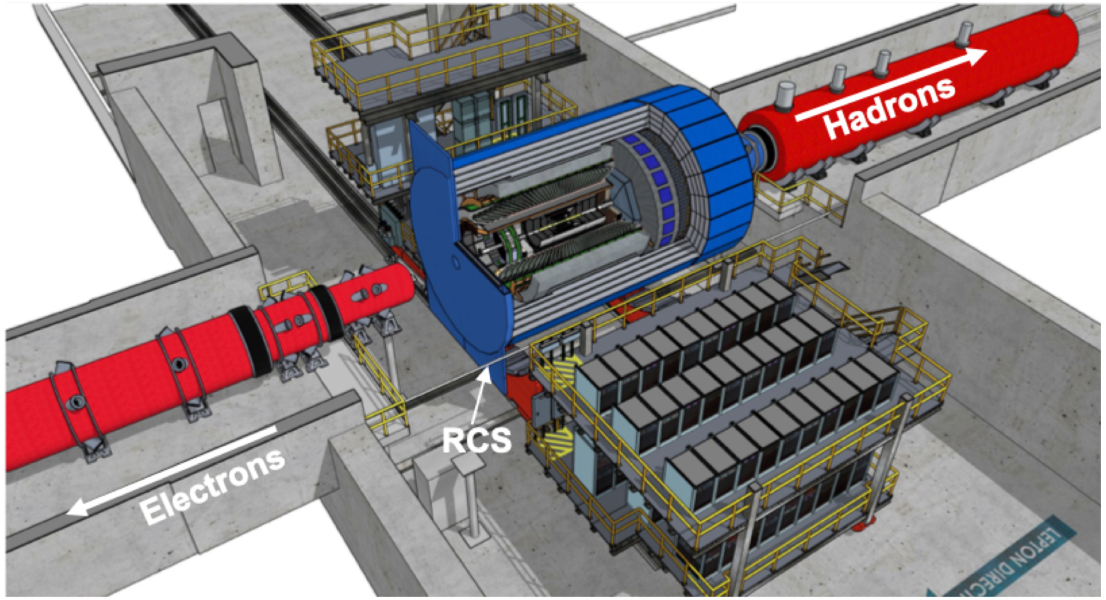
2022–2023: Formation of ePIC Collaboration

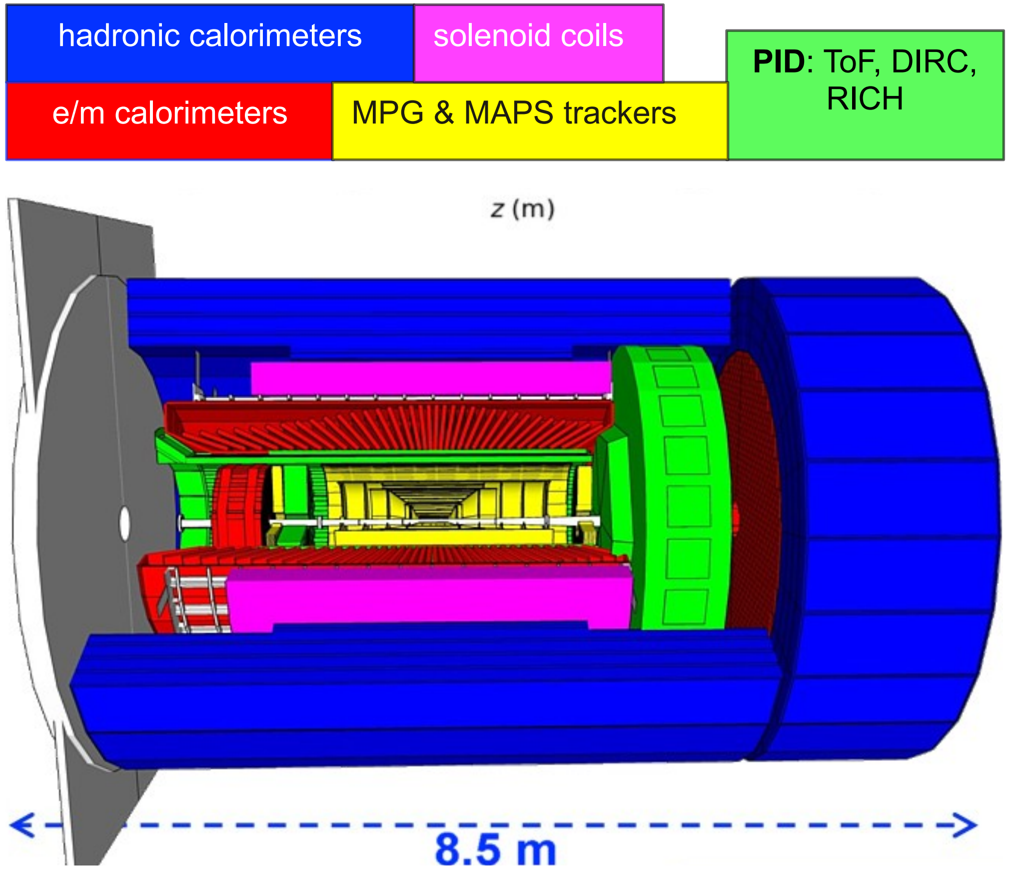


ePIC Collaboration Meeting at Jefferson Lab in January 2023

General Purpose Detector for ePIC

Integrated interaction and detector region (+/- 40 m) to get ~100% acceptance for all final state particles, and measure them with good resolution.





Overall detector requirements:

- Large rapidity (-4 < h < 4) coverage; and far beyond in far-forward detector regions.
- Large acceptance solenoid of 1.7 T (up to 2 T).
- High control of systematics: luminosity monitor, electron and hadron polarimetry.

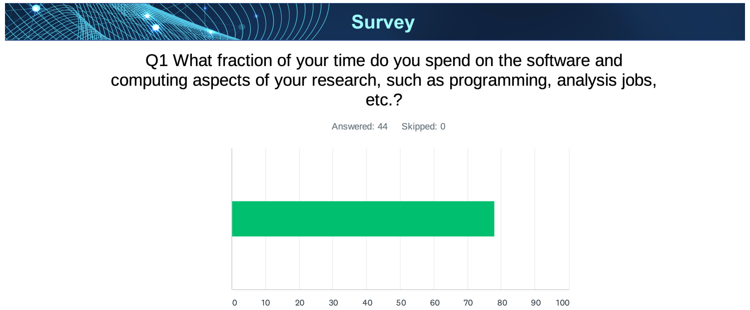
Lessons Learned About EIC Software

2016 – 2020	EIC Software Consortium (ESC)
2018 – now	Software Working Group (SWG) in EIC User Group (EICUG)
2019 – 2021	Yellow Report Initiative
2021 – 2022	Detector Collaboration Proposals
2022 – now	ePIC Collaboration
2016 – now	Workshop Series on Future Trends in Nuclear Physics Computing
2016 – now	Software & Computing Round Table

Our Vision for Software & Computing at the EIC

“The purpose of computing is insight, not numbers.” Richard Hamming (1962)

Software & computing are an integral part of our research:



Survey among NP Ph.D. students and postdocs in preparation of ”Future Trends in NP Computing”

- **Goal** We would like to ensure that scientists of all levels worldwide can participate in EIC analysis actively.
- **User-Centered Design**: To achieve this goal, we must engage the wider community in the development.

Rapid turnaround of data for the physics analysis and to start the work on publications:

- **Goal**: Analysis-ready data from the DAQ system.
- **Compute-detector integration** using streaming readout, AI/ML, and heterogeneous computing.

Software is in a very early life stage.

Common software projects based on Expression of Interest for EIC Software by wider community:

- Avoid duplication of the effort, e.g., workflows for distributed computing.
- Team up on challenges, e.g., running on heterogeneous computing resources.

Major Initiatives:

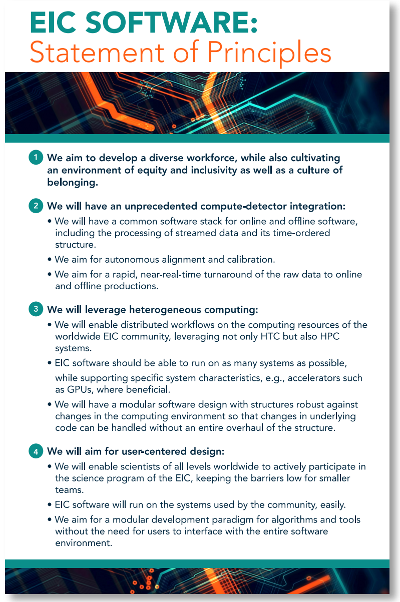
- Yellow Report: Physics case, the resulting detector requirements, and the evolving detector concepts for the EIC:
  - Mainly fast simulations and full simulations of detector components.
  - Foundation for detector collaboration proposals.

- Detector Collaboration Proposals: Very successful in large-scale, detailed full detector simulations:
  - ATHENA successfully developed a modular software stack based on common NHEP software.
  - ECCE successfully leveraged familiar software.
  - "State of Software" surveys: Commonality! One software stack!
  - "Lessons Learned" meetings organized EICUG to identify commonality between ATHENA and ECCE and proceed with work one software stack.

One Software Stack for the EIC

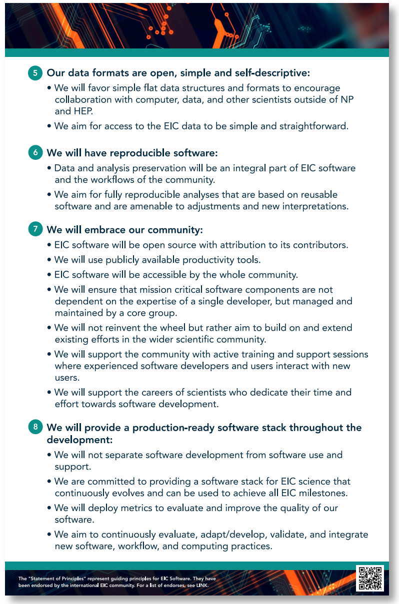
- **How to decide on our software stack?**
  - How do we ensure we work towards to our vision for EIC Software?
  - How do we ensure we meet the needs of the EIC community?

- **Solution: Statement of Principles**
  - Community process to define guiding principles for EIC Software.
  - Guiding principles define the requirements for EIC Software.
  - Endorsement by the international EIC community.



EIC SOFTWARE:
Statement of Principles

1. We aim to develop a diverse workforce, while also cultivating an environment of equity and inclusivity as well as a culture of belonging.
2. We will have an unprecedented compute-detector integration:
   - We will have a common software stack for online and offline software, including the processing of streamed data and its time-ordered structure.
   - We aim for autonomous alignment and calibration.
   - We aim for a rapid, near-real-time turnaround of the raw data to online and offline productions.
3. We will leverage heterogeneous computing:
   - We will enable distributed workflows on the computing resources of the worldwide EIC community, leveraging not only HTC but also HPC systems.
   - EIC software should be able to run on as many systems as possible, while supporting specific system characteristics, e.g., accelerators such as GPUs, where beneficial.
   - We will have a modular software design with structures robust against changes in the computing environment so that changes in underlying code can be handled without an entire overhaul of the structure.
4. We will aim for user-centered design:
   - We will enable scientists of all levels worldwide to actively participate in the science program of the EIC, keeping the barriers low for smaller teams.
   - EIC software will run on the systems used by the community, easily.
   - We aim for a modular development paradigm for algorithms and tools without the need for users to interface with the entire software environment.



5. Our data formats are open, simple and self-descriptive:
   - We will favor simple flat data structures and formats to encourage collaboration with computer, data, and other scientists outside of NP and HEP.
   - We aim for access to the EIC data to be simple and straightforward.
6. We will have reproducible software:
   - Data and analysis preservation will be an integral part of EIC software and the workflows of the community.
   - We aim for fully reproducible analyses that are based on reusable software and are amenable to adjustments and new interpretations.
7. We will embrace our community:
   - EIC software will be open source with attribution to its contributors.
   - We will use publicly available productivity tools.
   - EIC software will be accessible by the whole community.
   - We will ensure that mission critical software components are not dependent on the expertise of a single developer, but managed and maintained by a core group.
   - We will not reinvent the wheel but rather aim to build on and extend existing efforts in the wider scientific community.
   - We will support the community with active training and support sessions where experienced software developers and users interact with new users.
   - We will support the careers of scientists who dedicate their time and effort towards software development.
8. We will provide a production-ready software stack throughout the development:
   - We will not separate software development from software use and support.
   - We are committed to providing a software stack for EIC science that continuously evolves and can be used to achieve all EIC milestones.
   - We will deploy metrics to evaluate and improve the quality of our software.
   - We aim to continuously evaluate, adapt/develop, validate, and integrate new software, workflow, and computing practices.

The "Statement of Principles" represent guiding principles for EIC Software. They have been endorsed by the international EIC community. For a list of endorsers, see LINK.

PDF version, Webpage

Principle 1: DE&I

EIC SOFTWARE:
Statement of Principles

1. We aim to develop a diverse workforce, while also cultivating an environment of equity and inclusivity as well as a culture of belonging.

Principle 2: Compute-Detector Integration

2 We will have an unprecedented compute-detector integration:

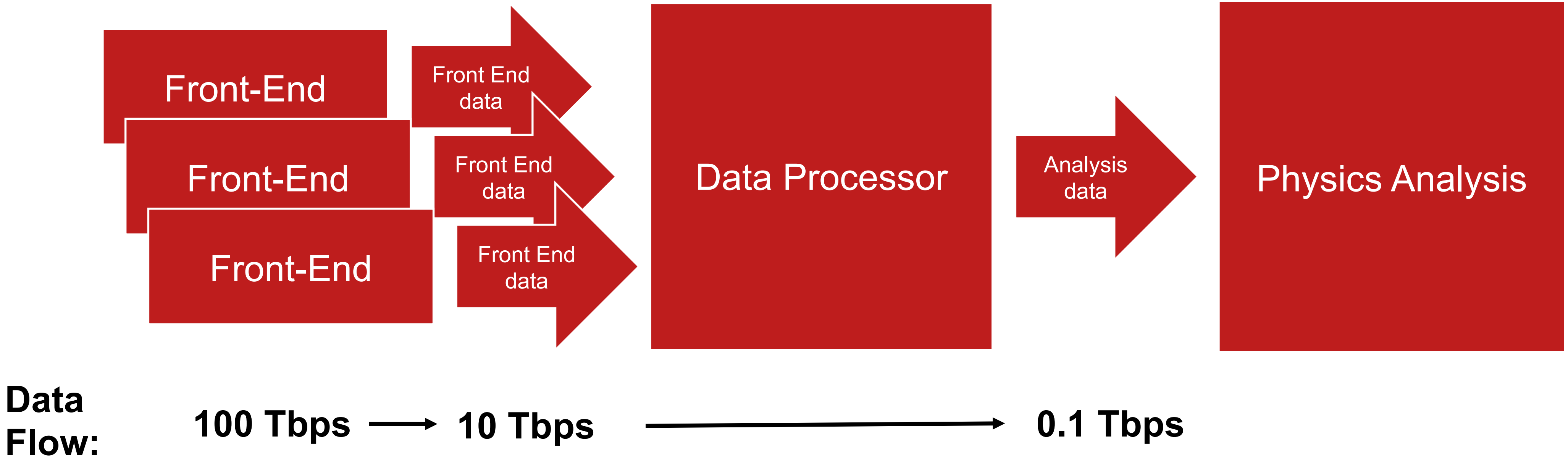
- We will have a common software stack for online and offline software, including the processing of streamed data and its time-ordered structure.

  Provisions for **streaming readout** from the start.

- We aim for autonomous alignment and calibration.
- We aim for a rapid, near-real-time turnaround of the raw data to online and offline productions.

Compute-Detector Integration to Maximize Science

- **Problem** Data for physics analyses and the resulting publications available after O(1year) due to complexity of NP experiments (and their organization).
  - Alignment and calibration of detector as well as reconstruction and validation of events time-consuming.
- **Goal** Rapid turnaround of data for physics analyses.
- **Solution** Compute-detector integration using:
  - AI/ML for autonomous alignment and calibration as well as reconstruction in near real time,
  - Streaming readout for continuous data flow and heterogeneous computing for acceleration.



Streaming Readout: Trigger-Less Data Acquisition

Definition of Streaming Readout

- Data is digitized at a fixed rate with thresholds and zero suppression applied locally.
- Data is read out in continuous parallel streams that are encoded with information about when and where the data was taken.
- Event building, filtering, monitoring, and other processing is deferred until the data is at rest in tiered storage.

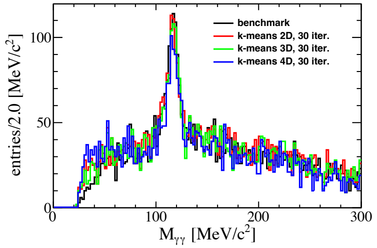
Advantages of Streaming Readout

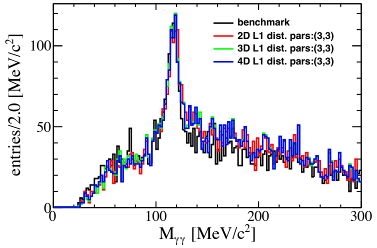
- Simplification of readout (no custom trigger hardware and firmware).
- Continuous data flow provides detailed knowledge of background.
- Streamline workflows and take advantage of other emerging technologies:
  - AI/ML for autonomous experimentation and control,
  - Heterogeneous computing.

On-Beam Validation of Streaming Readout at Jefferson Lab

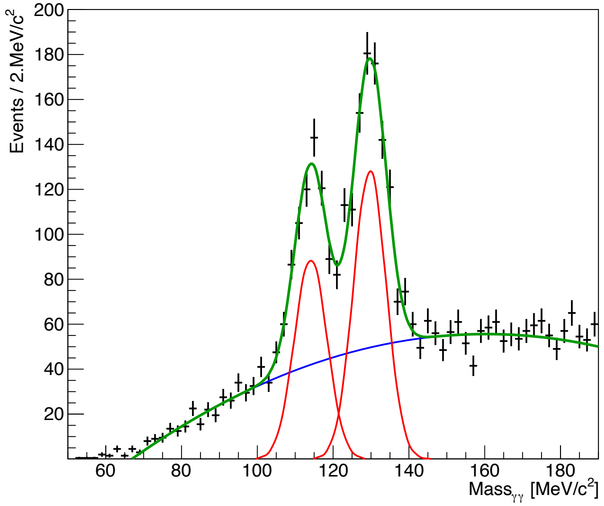
Tests included AI-supported real-time tagging and selection algorithms (*Eur.Phys.J.Plus* 137 (2022) 8, 958)

- Standard operation of **Hall-B CLAS12** with high-intensity electron-beam
- Streaming readout of forward tagger calorimeter and hodoscope
- Measurement of inclusive π^0 hadronproduction





- Prototype of EIC PbWO4 crystal EMCAL in **Hall-D Pair Spectrometer**
- Calorimeter energy resolution of SRQ compatible with triggered DAQ.



Principle 3: Heterogeneous Computing

3 We will leverage heterogeneous computing:

- We will enable distributed workflows on the computing resources of the worldwide EIC community, leveraging not only HTC but also HPC systems.

Software design should not limit what systems we can run on.

- EIC software should be able to run on as many systems as possible, while supporting specific system characteristics, e.g., accelerators such as GPUs, where beneficial.
- We will have a modular software design with structures robust against changes in the computing environment so that changes in underlying code can be handled without an entire overhaul of the structure.

Our design should be resilient against changing requirements, which we can accomplish by building a toolkit of orthogonal components.

Principle 4: User-Centered Design

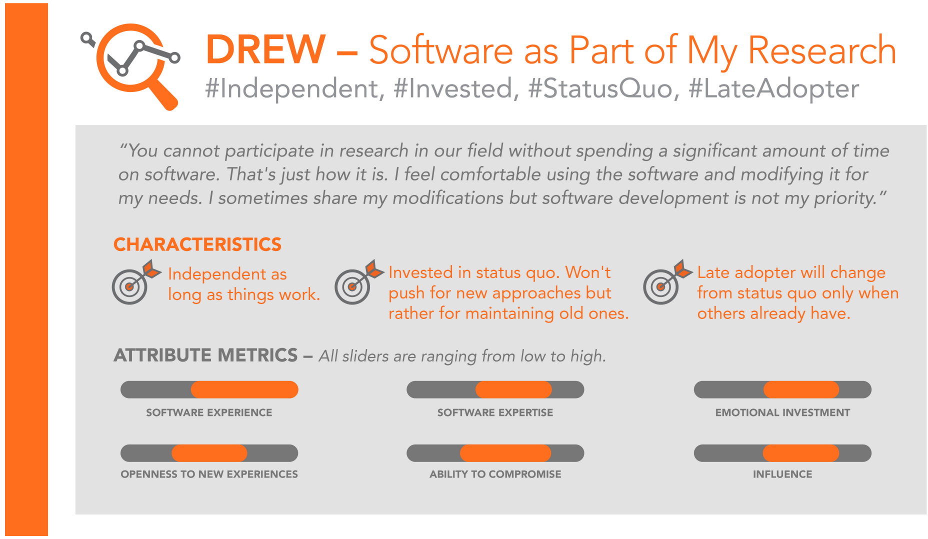
4 We will aim for user-centered design:

- We will enable scientists of all levels worldwide to actively participate in the science program of the EIC, keeping the barriers low for smaller teams.
- EIC software will run on the systems used by the community, easily.
- We aim for a modular development paradigm for algorithms and tools without the need for users to interface with the entire software environment.

Users should not need to know the entire toolchain to make meaningful contributions to a single component.

User-Centered Design

- **State of Software Survey**: Collected information on software tools and practices during the Yellow Report Initiative.
- As part of the State of Software Survey, we asked for volunteers for focus-group discussions:
  - Students (2f, 2m), Junior Postdocs (2f, 3m), Senior Postdocs (2f, 3m), Professors (5m), Staff Scientists (2f, 3m), Industry (2f, 2m)
- **Results from the six focus-group discussions**:
  - Extremely valuable feedback, documented many suggestions and ideas.
  - Developed user archetypes with Communication Office at Jefferson Lab and UX Design Consultant:



User Archetypes: Input to software developers as to which users they are writing software for:

- Software is not my strong suit.
- Software as a necessary tool.
- **Software as part of my research.**
- Software is a social activity.
- Software emperors.

- Repeated State of Software Survey after detector collaboration proposals:
  - The regular software census will be essential to better understand and quantify software usage throughout the EIC community. During the next survey, we will also ask on feedback on the user archetypes.

User-Centered Design: Listen to Users, and/then Develop Software

User-Centered Design
- Software census
- Focus groups and user archetypes
- Develop testing community

Discoverable Software
- Single point of entry
- Feasible option for >80% of EIC simulations and analyses
- Spack as package manager

Workflows
- Template repositories for key analyses
- Template repositories for validation workflows

Data and Analysis Preservations
- User analysis code/software registry
- Tutorials on reproducible analyses

Principle 5: Open, Simple, and Self-Descriptive Data Formats

5 Our data formats are open, simple and self-descriptive:

- We will favor simple flat data structures and formats to encourage collaboration with computer, data, and other scientists outside of NP and HEP.
- We aim for access to the EIC data to be simple and straightforward.

Principle 6: Reproducible Software

Data and analysis preservation is a hard problem, rarely effectively addressed. We will consider this from the start.

6 **We will have reproducible software:**

- Data and analysis preservation will be an integral part of EIC software and the workflows of the community.
- We aim for fully reproducible analyses that are based on reusable software and are amenable to adjustments and new interpretations.

Principle 7: Community

7 We will embrace our community:

- EIC software will be open source with attribution to its contributors.
- We will use publicly available productivity tools.
- EIC software will be accessible by the whole community.
- We will ensure that mission critical software components are not dependent on the expertise of a single developer, but managed and maintained by a core group.
- We will not reinvent the wheel but rather aim to build on and extend existing efforts in the wider scientific community. Focus on actual content.
- We will support the community with active training and support sessions where experienced software developers and users interact with new users.
- We will support the careers of scientists who dedicate their time and effort towards software development.

Principle 8: Development and Operation

8 We will provide a production-ready software stack throughout the development:

- We will not separate software development from software use and support.
- We are committed to providing a software stack for EIC science that continuously evolves and can be used to achieve all EIC milestones.
- We will deploy metrics to evaluate and improve the quality of our software.
- We aim to continuously evaluate, adapt/develop, validate, and integrate new software, workflow, and computing practices.

We have deliverables for each of the CD milestones. We will ensure our new development goes hand-in-hand with continuous reliability to ensure the EIC detector and its science program are successful. Our modular approach will facilitate controlled and reproducible incrementalism.

ePIC Software for the Realization of the ePIC Experiment

Our software design is based on **lessons learned in the worldwide NP and HEP community** and a **decision-making process** involving the whole community. We will continue to work with the worldwide NP and HEP community.

Modular Simulation, Reconstruction, and Analysis Toolkit using tools from the NP-HEP community

- MC Event Generators
- Detector Simulations in Geant4
- Readout Simulation (Digitization)
- Reconstruction in JANA2
- Physics Analyses

Edm4eic data model based on edm4hep and podio.
Geometry Description and Detector Interface using DD4hep.

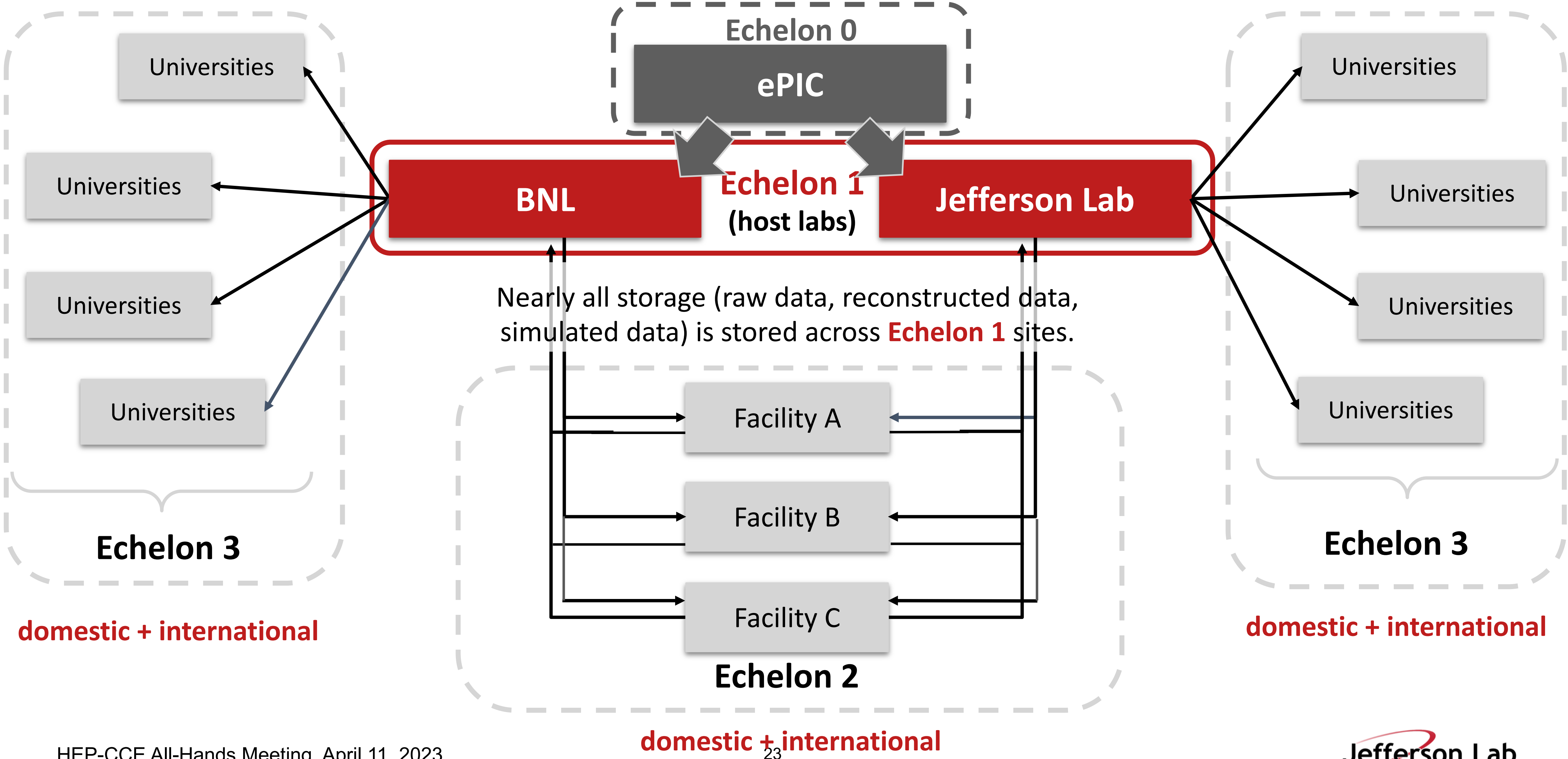
Continuous Integration for Detector and Physics Benchmarks and Reproducibility

We are providing a production-ready software stack throughout the development:

- **Milestone**: Software enabled first large-scale simulation campaign for ePIC.

We have a good foundation to meet the near-term and long-term software needs for ePIC.

Distributed Computing Model



Software & Computing is Ultimately About Science

We need to work together on great software for great science, on a global scale and with other fields:

User-Centered Design:

- We will **enable scientists of all levels worldwide** to actively participate in the EIC science program, keeping the barriers low for smaller teams.
- We will engage the international community in our design and development.

Next-Generation Simulations and Analysis Tools for High-Precision Measurements:

- We will collaborate with the international community on the **accurate modeling of physics and background processes** as well as their interplay with the ePIC detector.
- We will **collaborate with the international community** on data science for the EIC.

Data and Analysis Preservation:

- The success of the EIC science program depends on **fully reproducible, re-usable, and re-interpretable analyses**.
- We will make steady progress with **data and analysis preservation**, building on the experience and expertise of the international community.

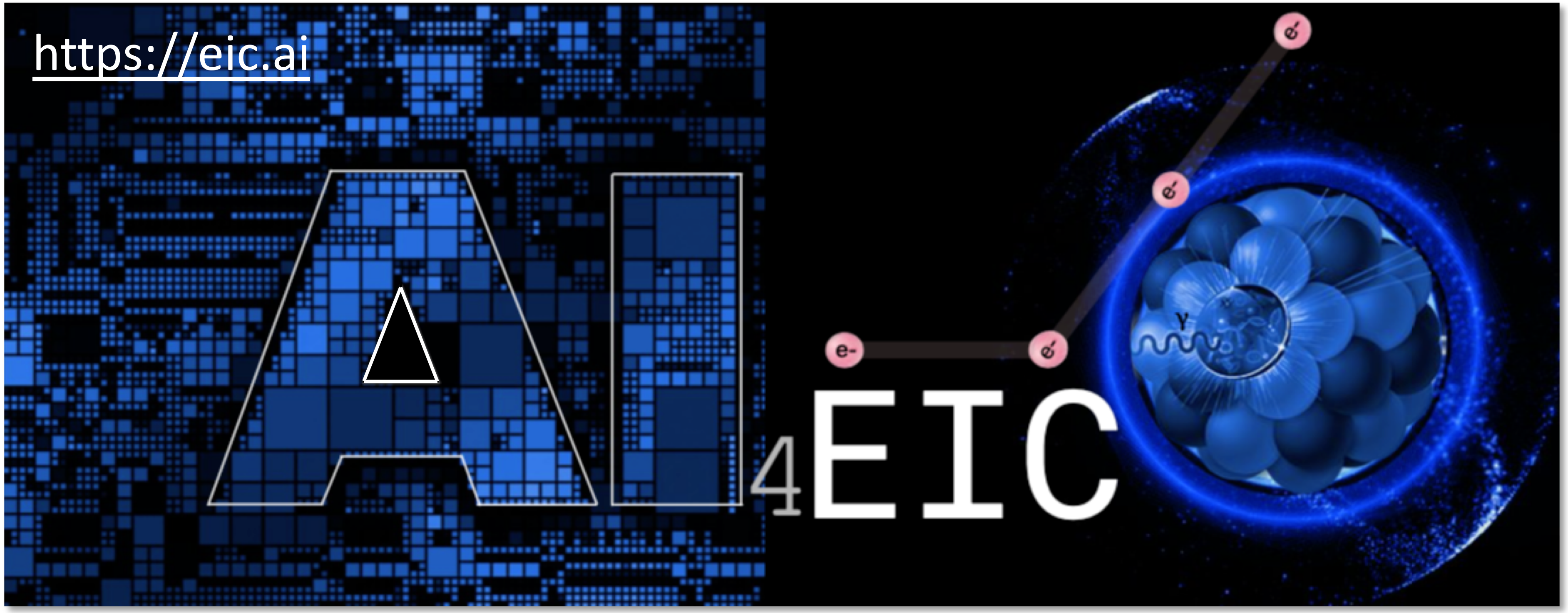
AI4EIC

AI/ML already has an **important presence in EIC** with many prototypes, e.g., for detector optimization or reconstruction methods using ML.

- Overview: [Colloquium: Machine learning in NP]
- **AI4EIC 2021 and 2022 workshops** with 200+ participants each

To explore and develop the full potential of **AI/ML for the EIC**, we as a community need to **move from prototyping to production** and add promising AI/ML solutions into our workflows.

- Promising candidate: Detector optimization using ML.



R&D Towards Next-Generation Detector Simulations

Detector Simulation

- **EIC focused project**
- **Turn-key application**
- **Built on top of Geant4 for full and fast simulations**
- **With library of potential detector option**

Requirements

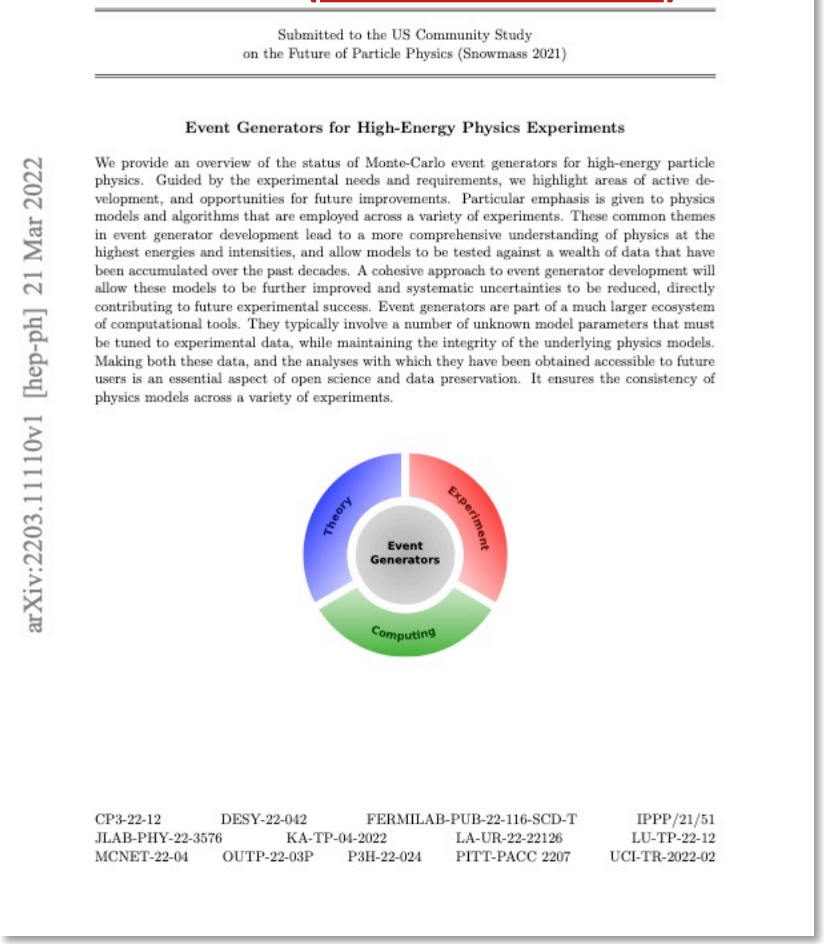
- Ease of **leveraging new and rapidly evolving** technologies:
  - AI/ML to accelerate simulations
  - Heterogeneous architectures:
    - AI/ML is the best near term prospect for using LCF/Exascale effectively.
- Ease of switching detector options
- Ease of switching between detailed and coarse detector descriptions

Project

- Support for high concurrency heterogeneous architectures and fast simulations integrated with full detector simulations allows to leverage AI/ML in Geant4.
- Next phase in concurrent Geant4: Sub-event parallelism.

MC4EIC

Cross-cutting aspects of MCEG R&D in NHEP (arXiv:2203.11110)



Submitted to the US Community Study
on the Future of Particle Physics (Snowmass 2021)

arXiv:2203.11110v1 [hep-ph] 21 Mar 2022

Event Generators for High-Energy Physics Experiments

We provide an overview of the status of Monte-Carlo event generators for high-energy particle physics. Guided by the experimental needs and requirements, we highlight areas of active development, and opportunities for future improvements. Particular emphasis is given to physics models and algorithms that are employed across a variety of experiments. These common themes in event generator development lead to a more comprehensive understanding of physics at the highest energies and intensities, and allow models to be tested against a wealth of data that have been accumulated over the past decades. A cohesive approach to event generator development will allow these models to be further improved and systematic uncertainties to be reduced, directly contributing to future experimental success. Event generators are part of a much larger ecosystem of computational tools. They typically involve a number of unknown model parameters that must be tuned to experimental data, while maintaining the integrity of the underlying physics models. Making both these data, and the analyses with which they have been obtained accessible to future users is an essential aspect of open science and data preservation. It ensures the consistency of physics models across a variety of experiments.

CP3-22-12 DESY-22-042 FERMILAB-PUB-22-116-SCD-T IPPP/21/51
JLAB-PHY-22-3576 KA-TP-04-2022 LA-UR-22-22126 LU-TP-22-12
MCNET-22-04 OUTP-22-03P P3H-22-024 PITT-PACC 2207 UCI-TR-2022-02

Monte Carlo Simulation of

- electron-proton (ep) collisions,
- electron-ion (eA) collisions, both light and heavy ions,
- including higher order QED and QCD effects,
- including a plethora of spin-dependent effects.

Common challenges, e.g. with DUNE or HL-LHC: **High-precision QCD measurements require high-precision simulations**.

Unique challenges MCEGs for electron-**ion** collisions and **spin-dependent** measurements, including novel QCD phenomena (e.g., GPDs or TMDs).
Will result in deeper understanding of QCD factorization and evolution, QED radiative corrections, hadronization models etc.

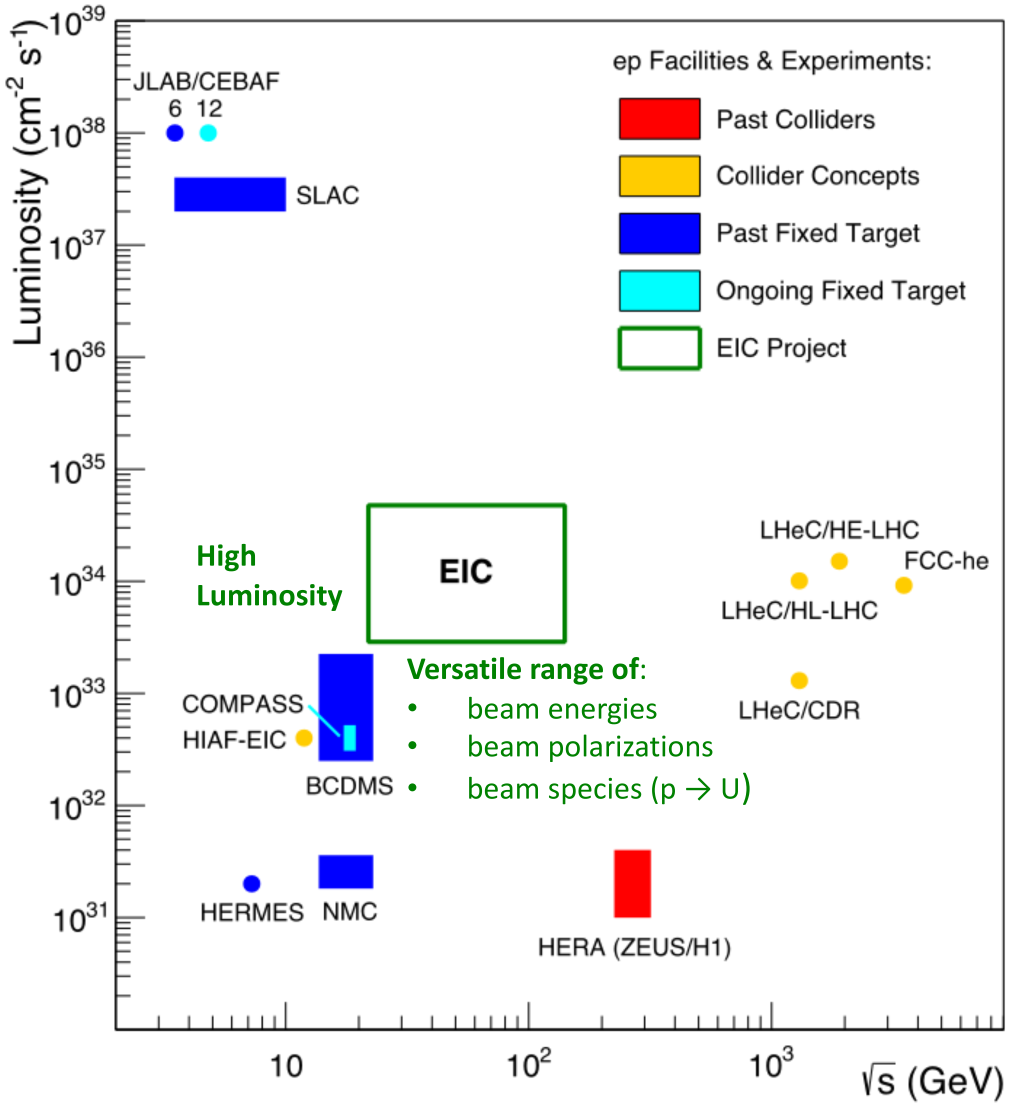
HEP-CCE Questionnaire

1) Have you updated your resource projections for the next decade? Have you identified "pain points" that require significant R&D?

We will be studying deep-inelastic scattering at high luminosity but low center-of-mass energies, as shown in on comparison slide **29**. In the final state, we expect either O(10) particles or one jet per interaction, which will occur every few bunch crossings.

We are currently developing a streaming computing model for ePIC, and we expect to have a better estimate of our resource needs once the model is fully developed. In the meantime, please refer to the early estimate from the detector collaboration proposals, which is shown on slide **30**.

EIC Science Parameters



Versatile range of

- Beam energies: $\sqrt{s}_{ep}$ range ~20 to ~100 GeV upgradable to ~140 GeV
- Beam polarizations for electrons, protons and light ions (longitudinal, transverse, tensor), at least ~70% polarization
- Ion beam species: D to heaviest stable nuclei

High luminosity

- 100 to 1000 times HERA luminosity

Raw Data Requirements

Estimates from ECCE Computing Plan

ECCE Runs	year-1	year-2	year-3
Luminosity	$10^{33}cm^{-2}s^{-1}$	$2 \times 10^{33}cm^{-2}s^{-1}$	$10^{34}cm^{-2}s^{-1}$
Weeks of Running	10	20	30
Operational efficiency	40%	50%	60%
Disk (temporary)	1.2PB	3.0PB	18.1PB
Disk (permanent)	0.4PB	2.4PB	20.6PB
Data Rate to Storage	6.7Gbps	16.7Gbps	100Gbps
Raw Data Storage (no duplicates)	4PB	20PB	181PB
Recon process time/core	5.4s/ev	5.4s/ev	5.4s/ev
Streaming-unpacked event size	33kB	33kB	33kB
Number of events produced	121 billion	605 billion	5,443 billion
Recon Storage	0.4PB	2PB	18PB
CPU-core hours (recon+calib)	191Mcore-hrs	953Mcore-hrs	8,573Mcore-hrs
2020-cores needed to process in 30 weeks	38k	189k	1,701k

HEP-CCE Questionnaire

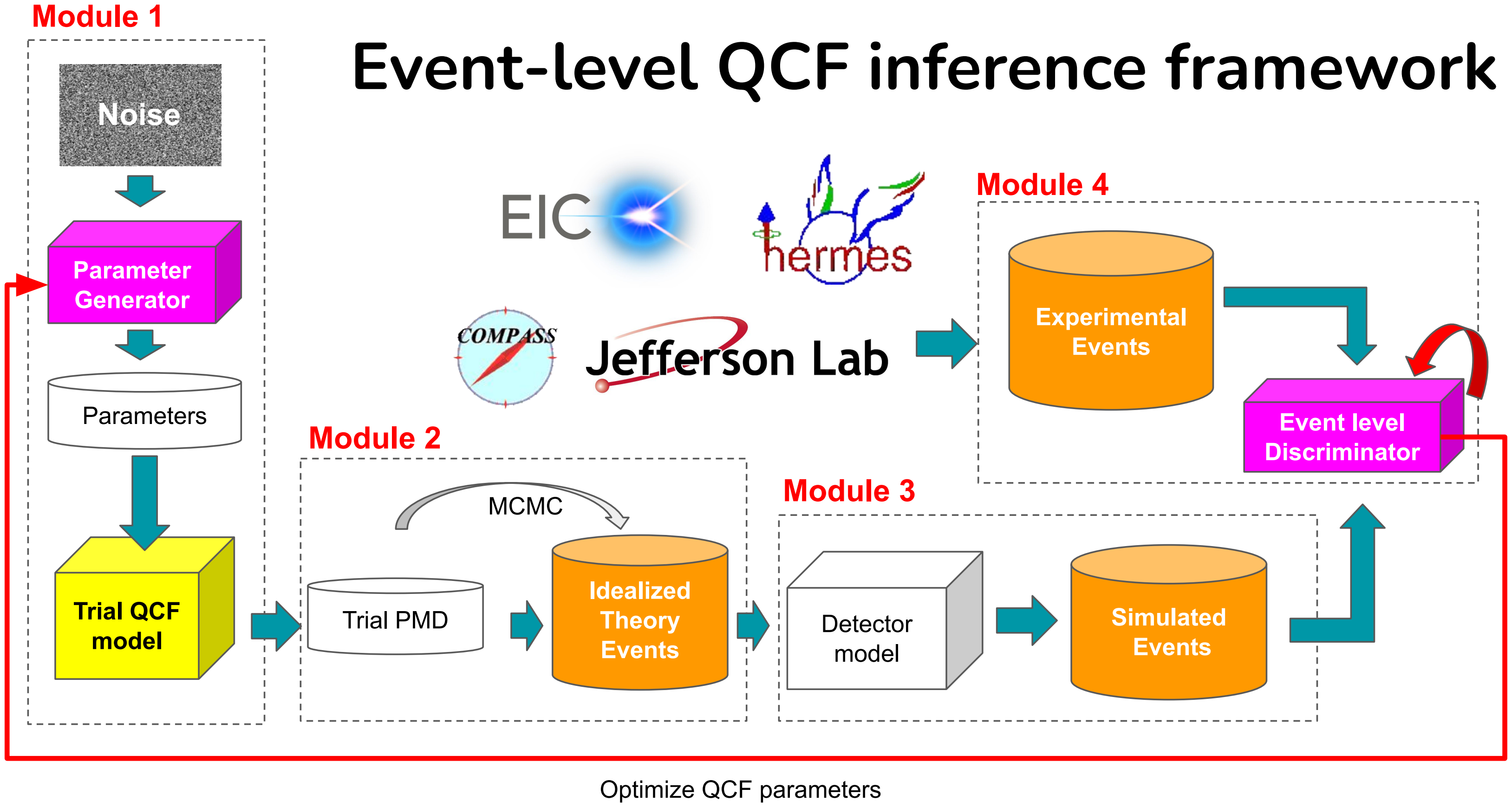
2) What is the role of HPC systems in your experiment today? Are you planning to increase this role in the future? Do you envision significant changes in your computing strategy that do not involve HPC?

We have utilized HPC systems for pattern recognition with high-granularity detectors (using ALCF/Polaris), and the related detector optimization. Currently, we envision using HPC systems aimed at burst level calculation, including AI/ML training, detector optimization, etc. We will also consider using HPC resources to aid in autonomous detector alignment and calibrations.

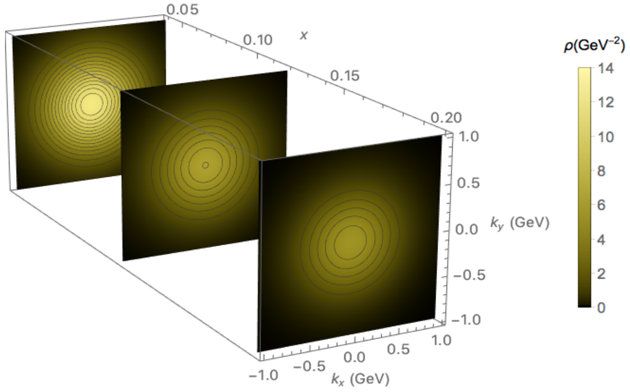
We plan to utilize both HPC and HTC resources for data processing and analysis for the EIC. It is difficult to predict the exact role that HPC will play in the future beyond its applications in AI/ML.

QuantOm - 3D Imaging of Quarks and Gluons using ML and Exascale

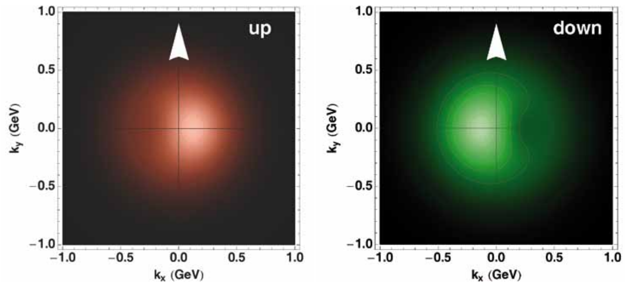
SciDAC project by ANL, Jefferson Lab, Virginia Tech



TMDs of unpolarized nucleon



TMDs of transversely polarized nucleon



HEP-CCE Questionnaire

3) If this particular issue is a concern, what is your strategy for increasing heterogeneity (ARM vs. x86, NVIDIA vs. AMD, FPGAs, ML accelerators)?

We currently have done exploratory work integration accelerators in our main simulation and reconstruction workflow. This includes abstraction layers such as alpaka to maintain flexibility. We also conducted initial studies optimizing topological clustering algorithms using SYCL.

We fully support x86 and ARM architectures in our software stack.

Part of this topic also ties in with the streaming computing model for ePIC, which is currently under active development in the collaboration.

HEP-CCE Questionnaire

4) What are the experiment priorities for algorithmic R&D (e.g., pattern recognition, simulation, generators, data management, and analysis pipelines), and what are the associated strategies (e.g., physics optimization, parallelization, ML, dedicated resources)?

- Accelerate the workflow for detector design optimization,
- Reconstruction for the holistic detector with strict modularity between the framework and its interfaces and the reconstruction algorithms,
- Reconstruction algorithms for particle identification, e.g., Cherenkov detectors,
- Reconstruction algorithms on event-level combining responses from various detector components,
- Automated validation of simulation.

HEP-CCE Questionnaire

5) What are the experiment strategies for I/O and storage optimization from physics-driven data reduction (e.g., nanoAODs) to lossy compression and network and storage-driven workflow optimization (e.g., dynamic replicas, tape vs. disk, intelligent networks)?

The strategy for the initial data reduction off the detector is closely tied in with the streaming computing model for ePIC, which is under active development.

Our event data model is based off EDM4hep using the PODIO toolkit.

We currently envision a multi-tiered set of reconstruction outputs tailored to specific observables and physics analyses.

The actual strategy for I/O and storage optimizations for ePIC are yet to be determined. We envision using distributed data storage powered by Rucio.

HEP-CCE Questionnaire

6) What else is important to know about your experiment computing needs?

We are at an early stage and would like to work together on great software for great science, on a global scale and with other fields.

Reach Out

ePIC Software & Computing Effort

Software and Computing Coordinator: Markus Diefenthaler
+ Deputy Coordinator **Operations: Wouter Deconinck**
+ Deputy Coordinator **Development: Sylvester Joosten**
+ Deputy Coordinator **Infrastructure: Torre Wenaus**

Electron-Ion Collider: Software & Computing

Markus Diefenthaler

mdiefent@jlab.org

- Software & Computing play an evergrowing role in modern science, including NP, HEP, and related fields. We will work together, globally and with other fields, to realize the science program for the EIC.
- We have a vision and guiding principles for EIC Software & Computing and laid the foundation of the ePIC software stack that enabled the first ePIC large-scale simulation campaign.
- **Many cross-cutting aspects with NHEP**:
  - AI/ML and heterogenous computing for next-generation simulations,
  - MCEGs development,
  - Streaming readout, AI/ML, heterogeneous computing for rapid turnaround of data for the physics analysis.