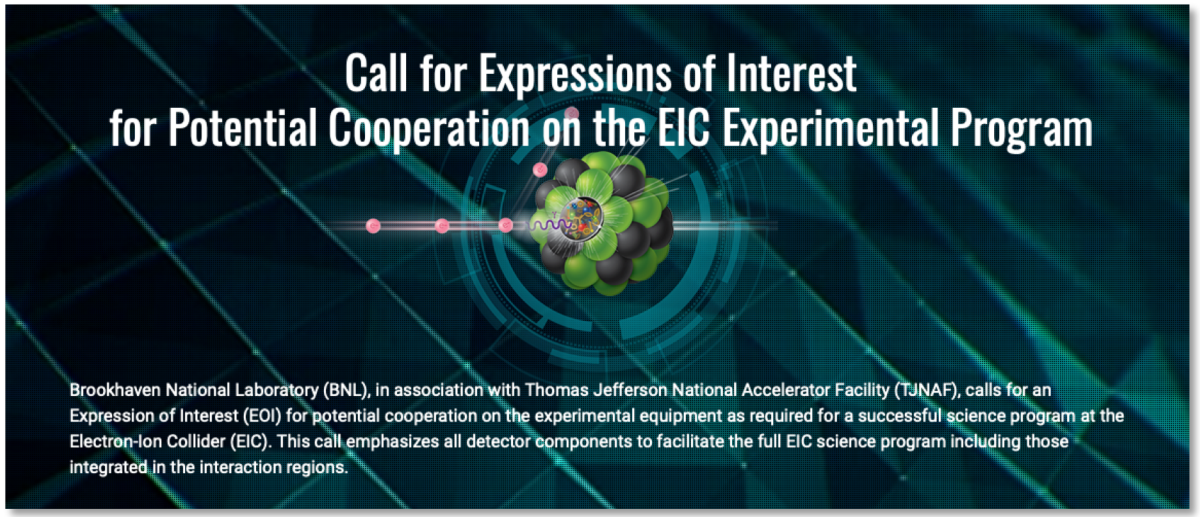
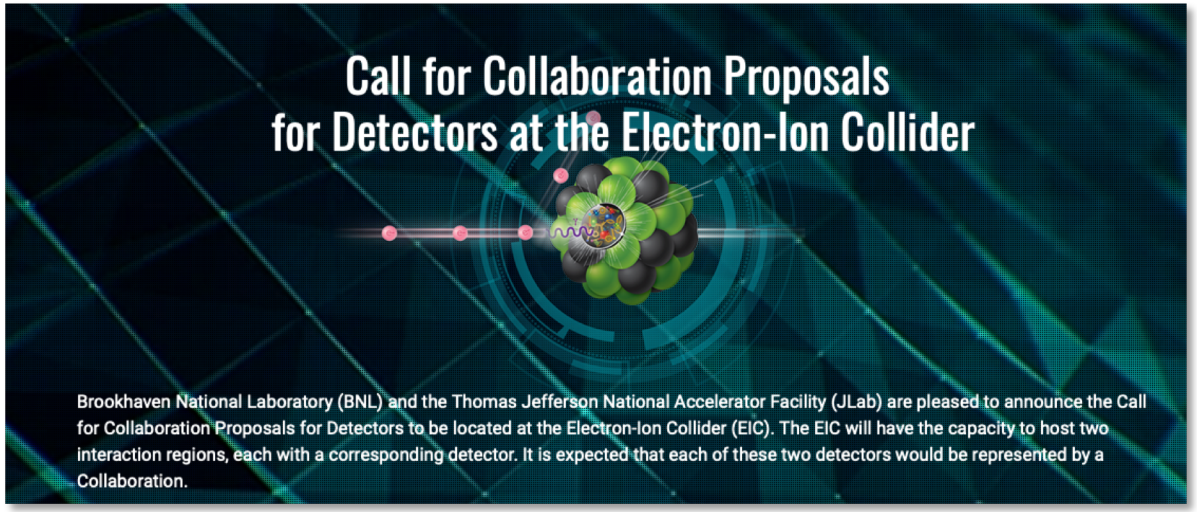
Software Working Group Priorities in 2021





Realize Software EoI

Support of Collaboration Proposals

Andrea Bressan (Trieste), Markus Diefenthaler (JLab), Torre Wenaus (BNL)

BROOKHAVEN NATIONAL LABORATORY

Jefferson Lab

UNIVERSITÀ DEGLI STUDI DI TRIESTE

EICUG Software Working Group

Electron-Ion Collider User Group

The world's most powerful microscope for studying the "glue" that binds the building blocks of visible matter.

HOME | JOIN EICUG | SCIENCE | ORGANIZATION | CALENDAR | SOFTWARE | DOCUMENTS | YELLOW REPORT | MEDIA | ADMIN

Home » EIC Software

EIC Software

Software Working Group

The Software Working Group (SWG) is open to all members of the EICUG to work on EICUG related software tasks. It communicates via its mailing list and organizes regular online and in-person meetings that enable broad and active participation from within the EICUG as a whole.

The SWG has participated in the call for Expressions of Interest (EoI) and intends to carry its EoI for Software forward as a living document that will evolve towards a work plan for the SWG, setting priorities for the next years and goals for the next decade.

In addition to that, it will support the work on the simulation efforts for the collaboration proposals for detectors at the EIC.

For questions about the Software Working Group, please contact the conveners (Andrea Bressan (Trieste), Markus Diefenthaler (JLab), and Torre Wenaus (BNL)).

Important links

Mailing list	eicug-software@eicug.org (subscribe via Google Group)
EIC organization on GitHub	https://github.com/eic
Website	https://eic.github.io

Copyright © 2021, Electron-Ion Collider User Group — Designed by Zymphonies

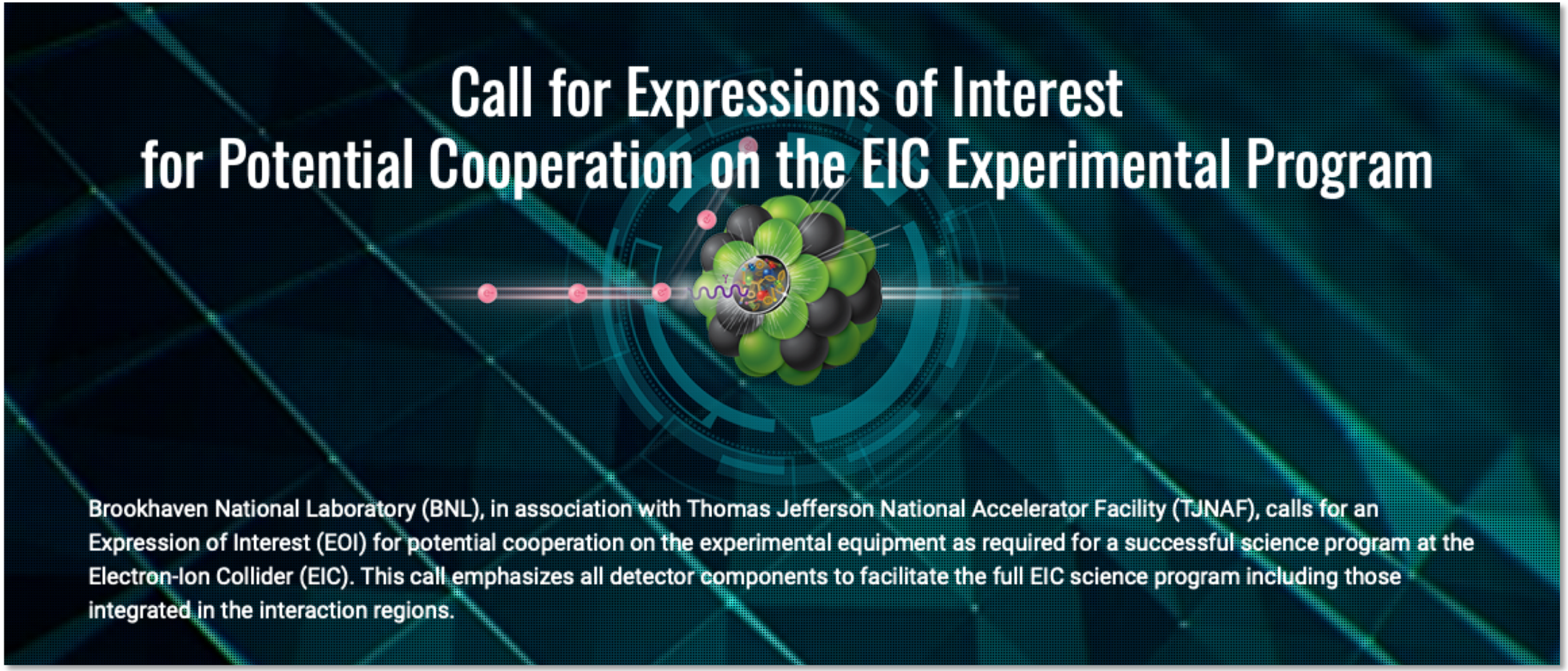
Unique software challenges for EIC Science

Scientific Problem Space

- Focus on non-perturbative QCD phenomena
- MC event generators for electron-**ion** collisions and **spin-dependent** measurements
  - including novel QCD phenomena (e.g., GPDs, TMDs, Wigner functions)
- **Analyses considering large number of signal events simultaneously** (or multiple times)
  - **Contrary** to separating a few events from a large number of background events
    - **Example** Search of rare events with novel topologies
  - **Example** complexity of multi-dimensional, strongly correlated relationships among data (e.g., GPDs, TMDs, Wigner functions)
  - **Example** high-precision results which require complex analyses to control systematic uncertainties
  - Require unique software and computing strategies

Work on EIC with high luminosity over a wide range of center-of-mass energies and highly polarized beams will **push R&D on these challenges**.

SWG Priority: Realize Software EoI



Community Input for Expression of Interest

Software Needs

Requirements What software needs for EIC Software would you like to highlight now, in a few years, and for the completion of the EIC project?

Technologies & Techniques What software technologies and techniques should be considered for the EIC?

Meeting Software Needs

What resources can your group contribute?

Expression of Interest for Software

Expression of Interest (EOI) for Software

Please indicate the name of the contact person for this submission:

Conveners of the Software Working Group:
- A. Bressan, M. Diefenthaler, and T. Wenaus
- eicug-software-conveners@eicug.org

Please indicate all institutions collectively involved in this submission of interest:

29 institutions

ANL	Argonne National Laboratory
BNL	Brookhaven National Laboratory
CEA/Irfu	IRFU at CEA /Saclay institute
EIC-India	Akal University, Central University of Karnataka, DAV College Chandigarh, Goa University, Indian Institute of Technology Bombay, Indian Institute of Technology Delhi, Indian Institute of Technology Indore, Indian Institute of Technology Patna, Indian Institute of Technology Madras, Malaviya National Institute of Technology Jaipur, Panjab University, Ramkrishna Mission Residential College Kolkata
IMP-CAS	Institute of Modern Physics - Chinese Academy of Sciences
INFN	Istituto Nazionale di Fisica Nucleare
JLab	Thomas Jefferson National Accelerator Facility
LANL	Los Alamos National Laboratory
LBNL and UC Berkeley	Lawrence Berkeley National Laboratory and University of California, Berkeley
NCBJ	National Centre for Nuclear Research
OhioU	Ohio University
ORNL	Oak Ridge National Laboratory
SBU	Stony Brook University
SLAC	SLAC National Accelerator Laboratory
SU	Shandong University

https://indico.bnl.gov/event/8552/contributions/43221/

Common Projects

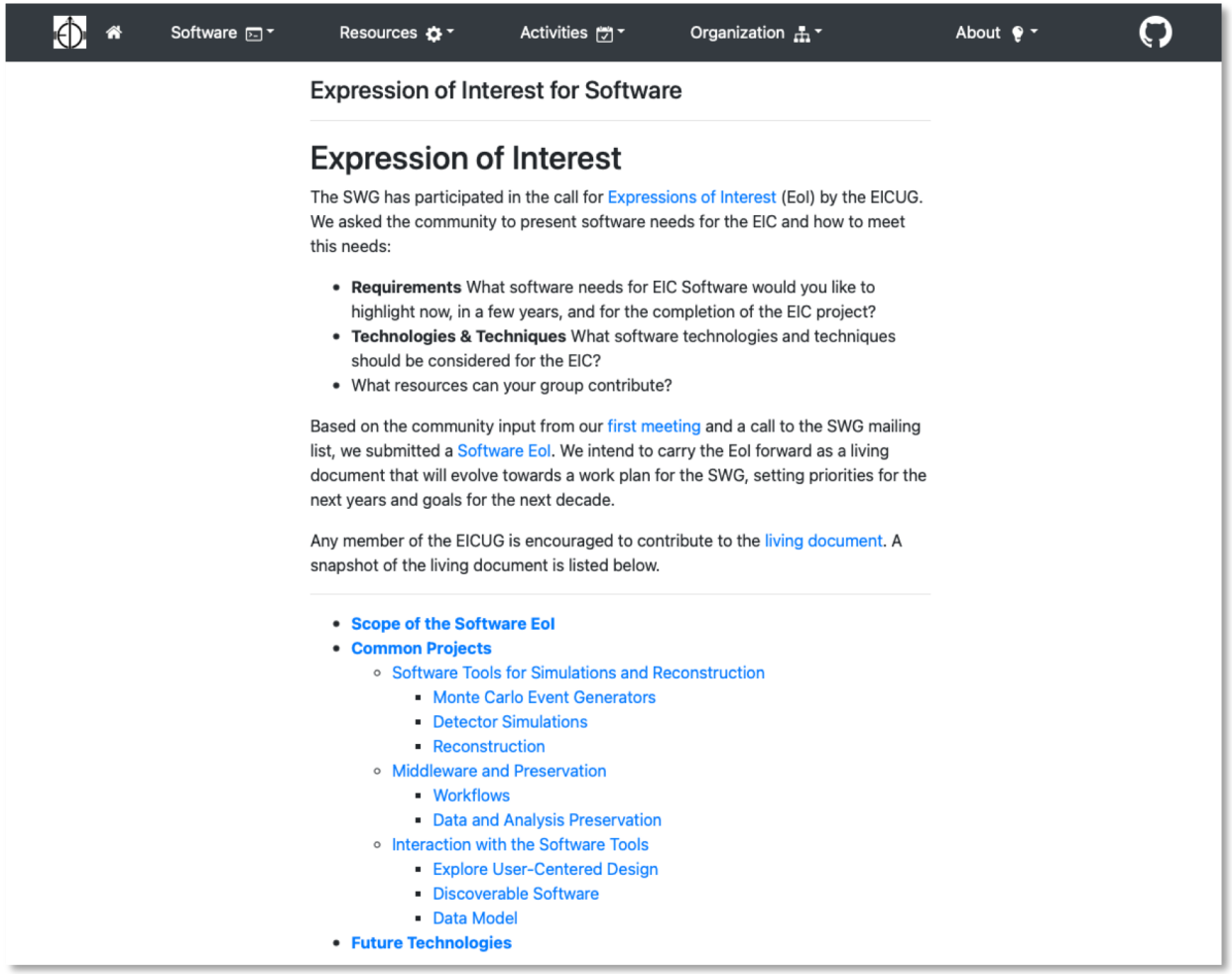
- **Software Tools for Simulations and Reconstruction**
  - Monte Carlo Event Generators
  - Detector Simulations
  - Reconstruction
  - **Validation**
- **Middleware and Preservation**
  - Workflows
  - Data and Analysis Preservation
- **Interaction with the Software Tools**
  - Explore User-Centered Design
  - Discoverable Software
  - Data Model

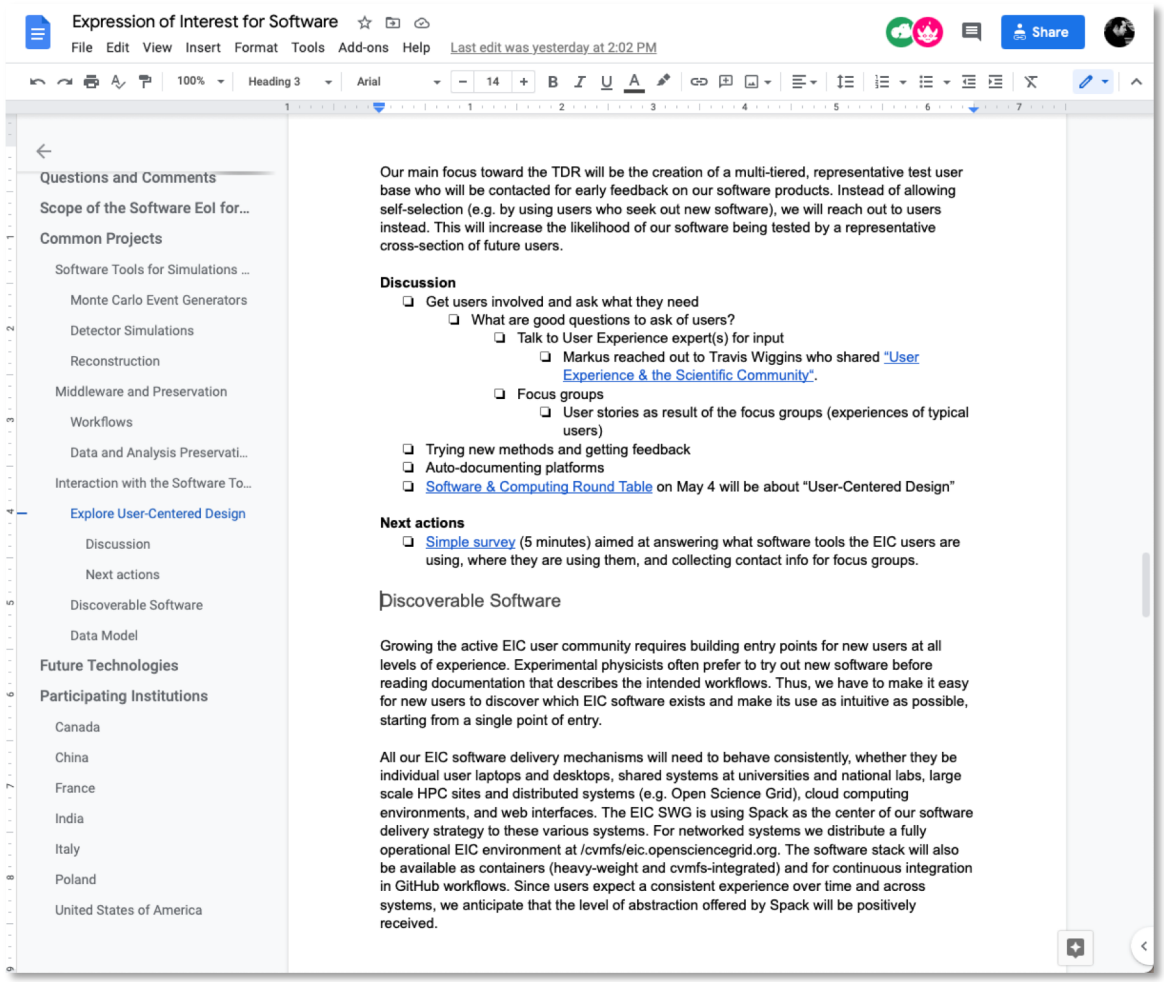
Future Technologies

- Artificial Intelligence
- Heterogeneous computing
- New languages and tools
- Collaborative software

EoI Evolving Towards Work Plan for the SWG

https://eic.github.io/activities/eoi.html



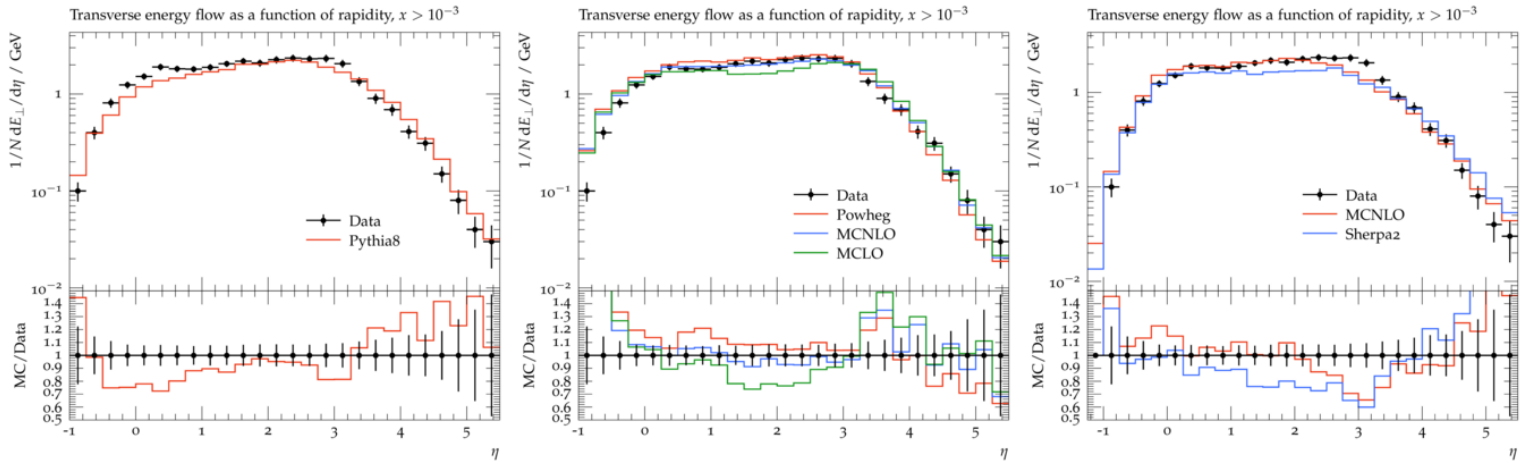


Example Project: Compare MCEGs Results with HERA Data

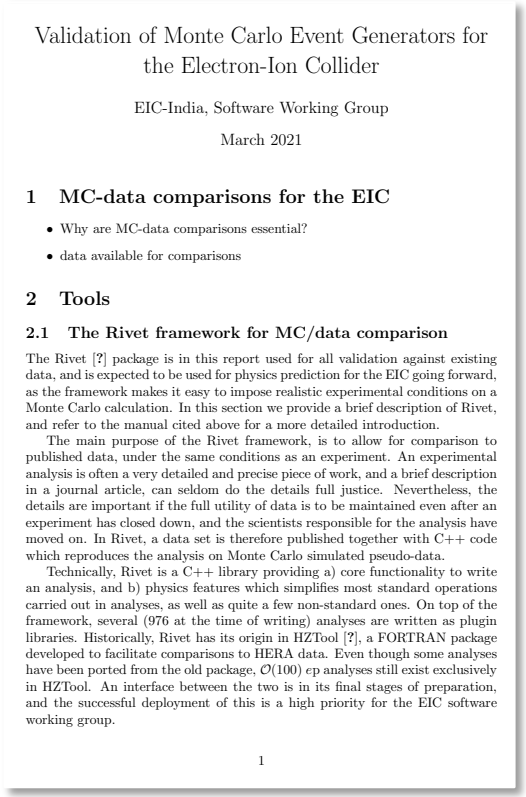
EIC-India, MCnet

Current activity

- Comparison to published results using RIVET and understand differences
- **Provide initial findings and results in a EICUG report**
  - Overview of where we stand in understanding HERA data with current physics and models implement in MCEGs



Pythia8 Herwig7 Sherpa2



Validation of Monte Carlo Event Generators for the Electron-Ion Collider

EIC-India, Software Working Group

March 2021

1 MC-data comparisons for the EIC

- Why are MC-data comparisons essential?
- data available for comparisons

2 Tools

2.1 The Rivet framework for MC/data comparison

The Rivet [?] package is in this report used for all validation against existing data, and is expected to be used for physics prediction for the EIC going forward, as the framework makes it easy to impose realistic experimental conditions on a Monte Carlo calculation. In this section we provide a brief description of Rivet, and refer to the manual cited above for a more detailed introduction.

The main purpose of the Rivet framework, is to allow for comparison to published data, under the same conditions as an experiment. An experimental analysis is often a very detailed and precise piece of work, and a brief description in a journal article, can seldom do the details full justice. Nevertheless, the details are important if the full utility of data is to be maintained even after an experiment has closed down, and the scientists responsible for the analysis have moved on. In Rivet, a data set is therefore published together with C++ code which reproduces the analysis on Monte Carlo simulated pseudo-data.

Technically, Rivet is a C++ library providing a) core functionality to write an analysis, and b) physics features which simplifies most standard operations carried out in analyses, as well as quite a few non-standard ones. On top of the framework, several (976 at the time of writing) analyses are written as plugin libraries. Historically, Rivet has its origin in HZTool [?], a FORTRAN package developed to facilitate comparisons to HERA data. Even though some analyses have been ported from the old package, $\mathcal{O}(100)$ ep analyses still exist exclusively in HZTool. An interface between the two is in its final stages of preparation, and the successful deployment of this is a high priority for the EIC software working group.

1

Example for EoI Planning (Work In Progress)

User-Centered Design

Q1 Survey of software practices in the EIC community ("what was used for YR")
Q2 EIC SW tutorials: follow-up survey
Q3 Develop user profiles

Discoverable Software

Q1 CI for spack, cvmfs/spack
Q2 Regular release of consistent environments on individual systems, cvmfs, as containers
Q2 Increase software provided on cvmfs
Q3 Become option for >80% of analyses

Workflows

Q1 Template repositories for key analyses
Q2 Cvmfs and containers, OSG job specs
Q3 REANA on BNL / JLab resources
Q4 Template repositories and workflows for validation

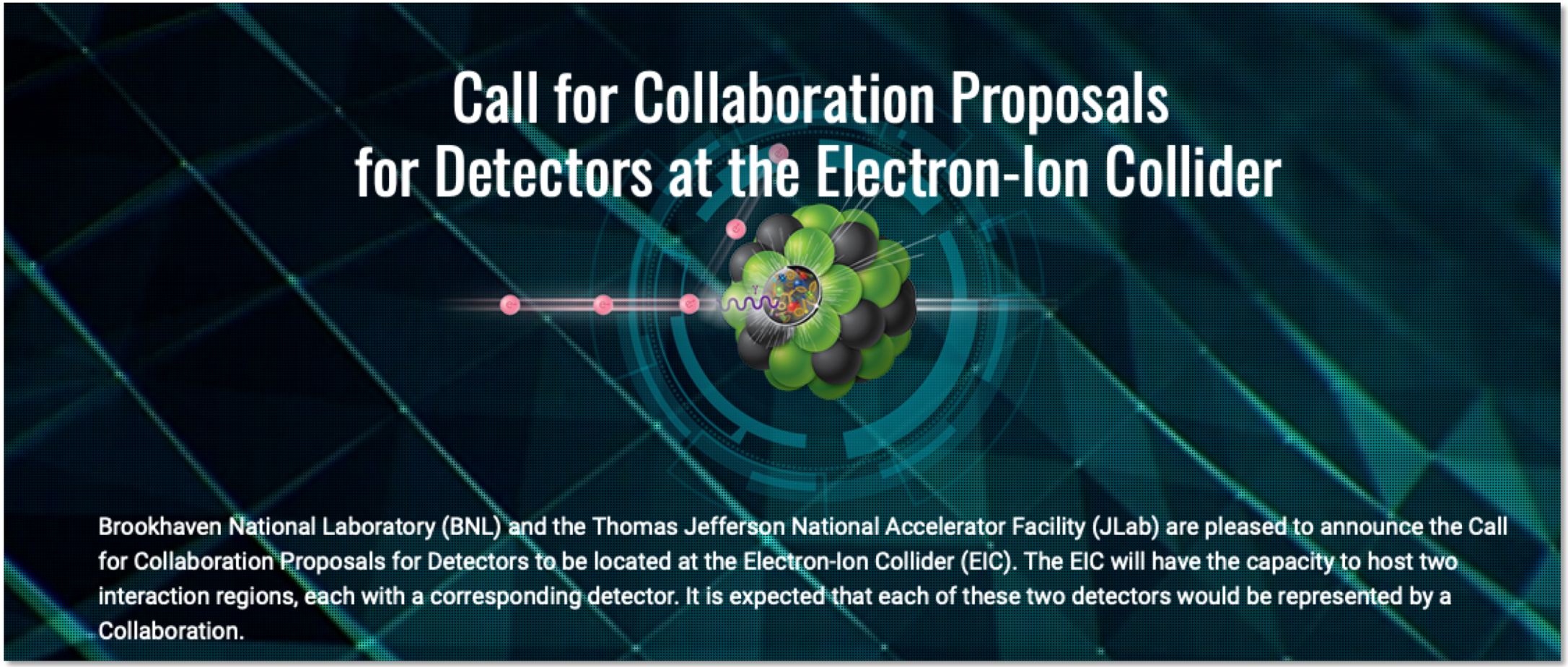
Data and Analysis Preservation

Q1 Policy framework for github/eic
Q2 User software/analysis code registry
Q2 Tutorials for analysis code on git
Q4 Tutorials on reproducible analysis

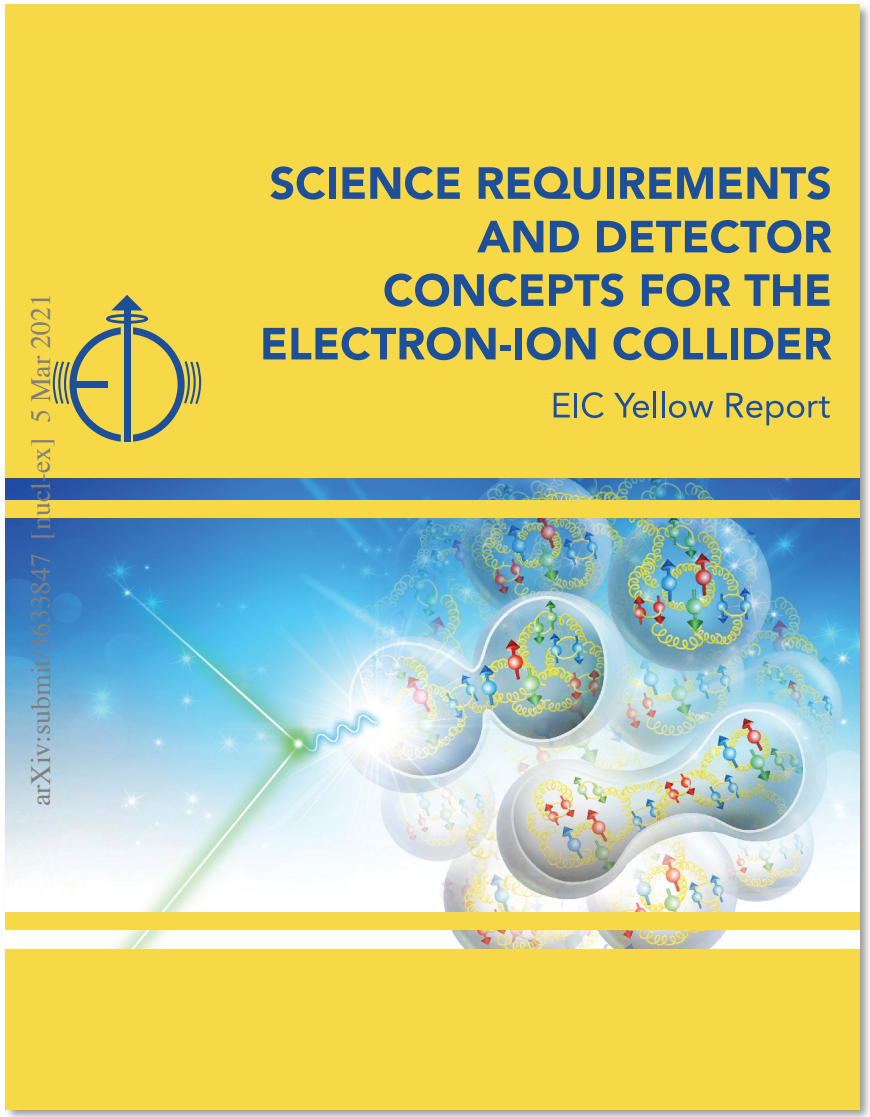
More on CI and Workflows for Validation by Whit

- ANL is part of the **Software EoI** and will share their experience with R&D on automated benchmarking workflows with the community (**Workflows and preservation**).

SWG Priority: Support of Collaboration Proposals



Support of the Yellow Report Initiative

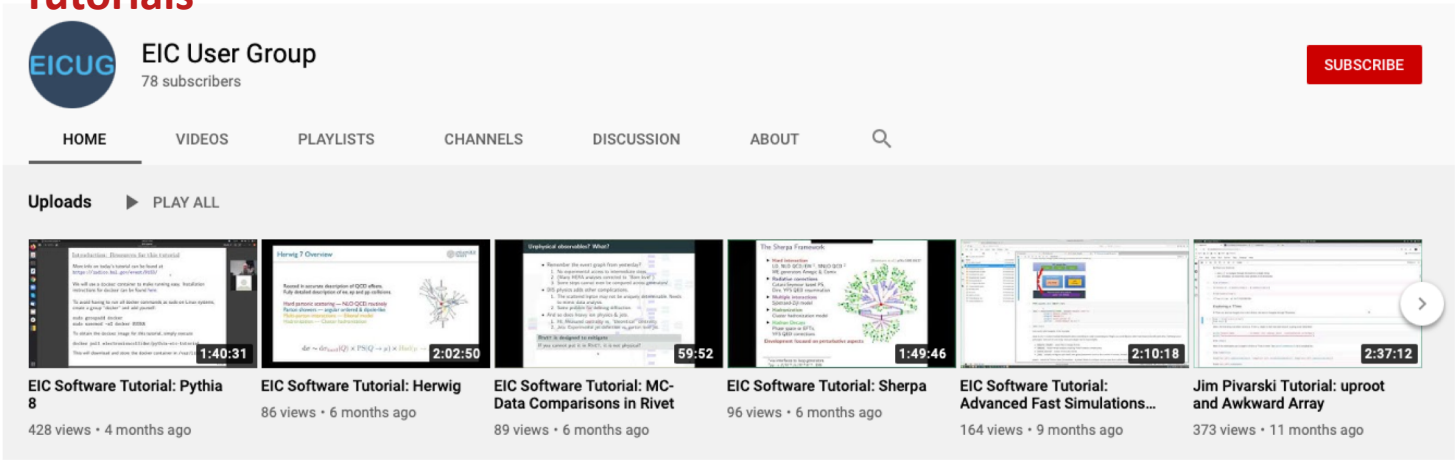


Software Collection

- **MCEGs**
- **Fast simulation tool** eic-smear, Delphes
- **Full simulation tool** EicRoot, ESCalate, Fun4All
- **Integration** EicToyModel

eic.github.io Main portal to EIC software, repositories, documentation and resources

Tutorials

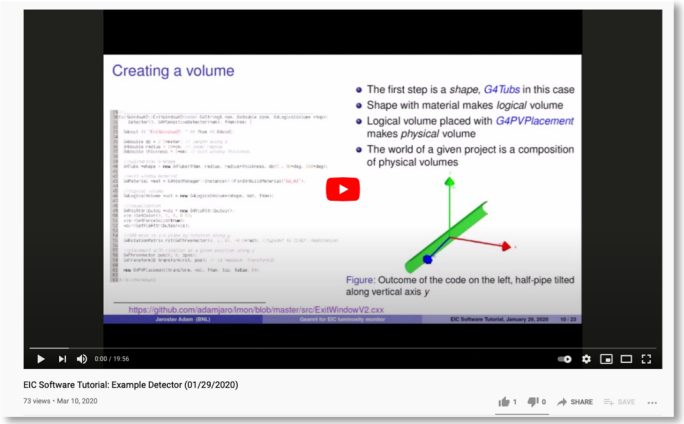


Support of Detector Full Simulations

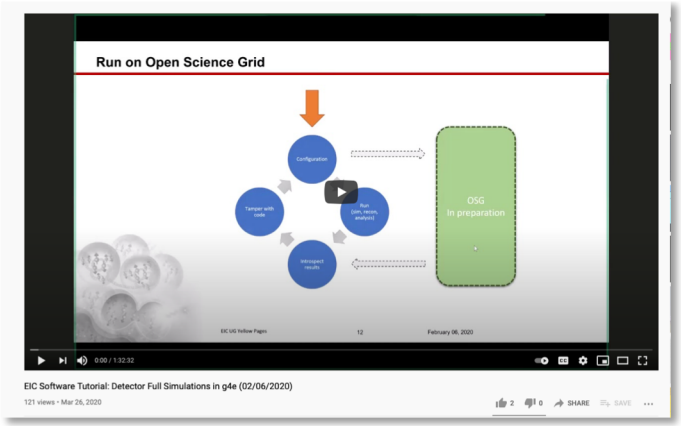
YouTube

- Tutorials on how to **integrate standalone Geant4 simulations** in ESCalate and Fun4All (January and February 2020)

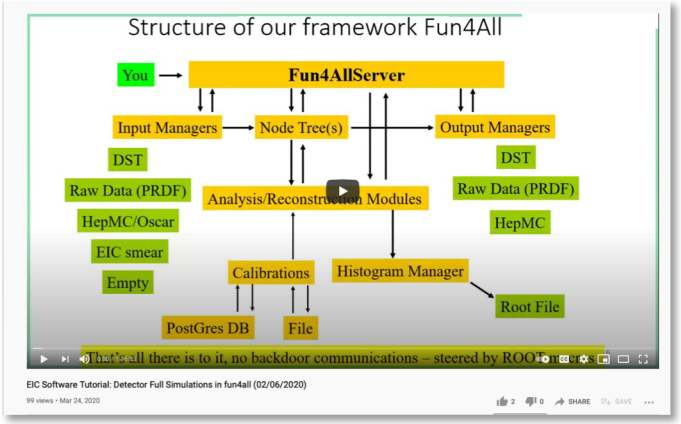
Example Detector



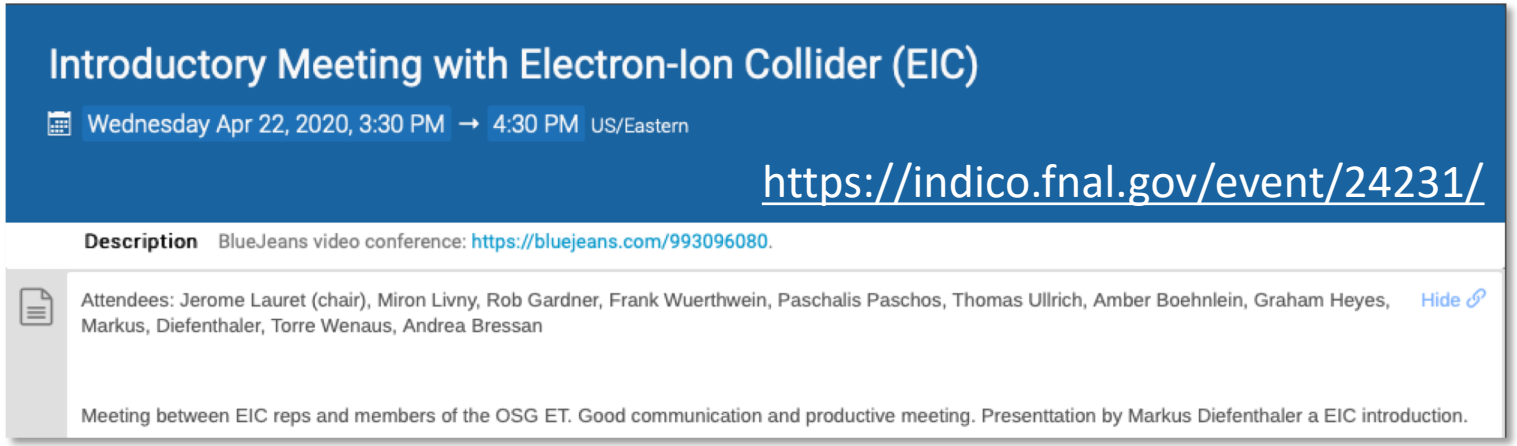
Integration in ESCalate (g4e)



Integration in Fun4All



- Work on **federated resources**



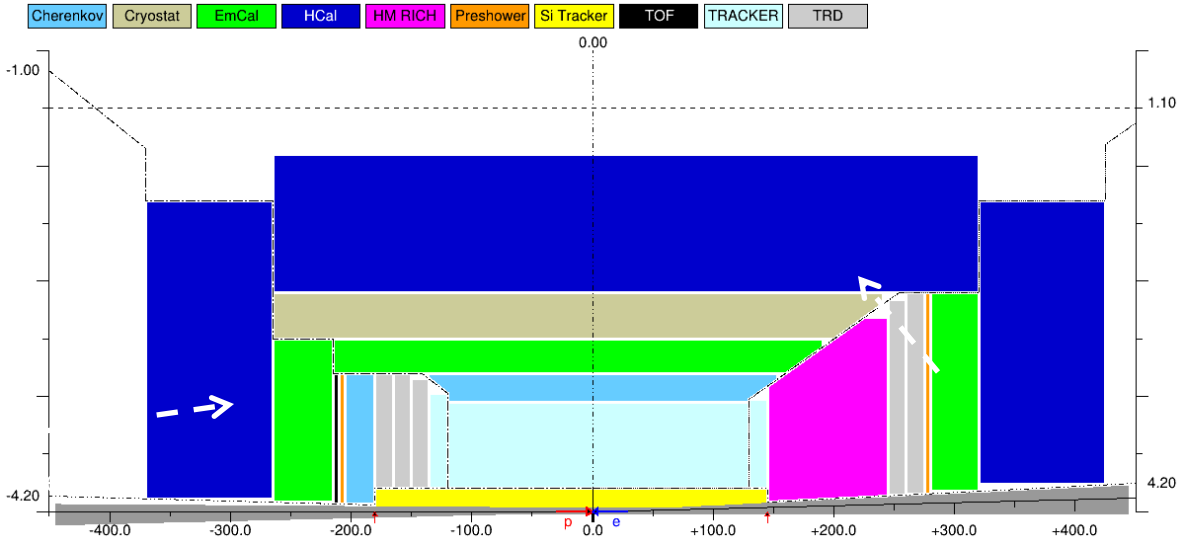
Support of Integration: EicToyModel

A. Kiselev

https://github.com/eic/EicToyModel

A tool to model & generate EIC Central Detector *templates* in a way

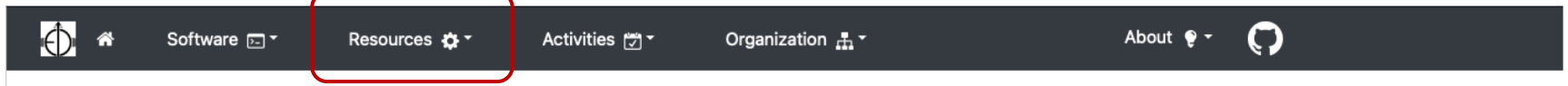
- new geometries (models) can be generated, *quickly* and *by everybody*
- represented instantly in a WYSIWYG fashion
  - the sub-detector *container objects* are guaranteed to not overlap either with each other or with the IR vacuum chamber elements
  - technically they can be imported in Geant4 frameworks in a consistent way and used as wrappers to the *real* sub-detectors
  - they can be exported in a CAD format to be used in the engineering design of the detector support structures and / or laying out services



EIC Computing

Coordinated by J. Lauret (BNL) and G. Heyes (JLab)

https://eic.github.io/resources/overview.html



Computing Resources available at BNL and JLab

BNL and JLab provide computing resources for the worldwide EIC community:

- **CPU**
- **Workload Management**
- **Storage**
  - Data Storage
  - Auxiliary Data Services
- **GitHub organization**

You can use your existing BNL or JLab account or register as new user at the host labs:

- Request a BNL computing account
- Request a JLab computing account (register as `User-remote` and state `EIC` as your sponsor)

CPU

BNL	JLab
Dedicated EIC resources: 7 machines with 64 core, 6×64 batch slots = 384 slots and one for interactive use (VMs). Each node has dual CPU AMD EPYC 7351.	JLab operates a common cluster. 25k job slots. Delivering ~ 76M core hours per year normalized to dual CPU AMD EPYC 7351.
1,000 slots allocated from a variety of CPUs (Intel® Xeon® and AMD EPYC)	EIC is currently one of ten projects sharing a 10% fair-share of the cluster. So guaranteed 7.6M core hours if the other nine are idle 0.8M if all are busy
31k slots available to all experiments that can be dynamically allocated upon priorities (358 M core hours per year).	A further 4MCH/yr dedicated to EIC will be added when demand requires it.

Software Working Group and Collaborations Proposals

Work with the EIC community

- EIC collaborations will determine for themselves what they do for software, but that will likely include **common software**.
- SWG supported Yellow Report Initiative and will **support call for detector collaborations proposals**:
  - **This includes of course the work on a detector concept for the 2nd IR.**
  - Simulation efforts by ECCE, EIC@IP6, and possible other initiatives can be part of SWG.
  - Common software projects by the SWG highly relevant for ECCE, EIC@IP6, and possible other initiatives.

Common Software

- Define requirements for EIC Software and **common software projects**:
  - Software needs of the EIC addressed in EoI.
  - Evolve with the EIC community and the EIC project.
- Work together on **common software projects** based on these requirements.
  - Avoid duplication of the effort, e.g., workflows for running on computing resources at the host labs or the OSG.
  - Team up on challenges, e.g., running on heterogenous resources.
- Continue to build a **EIC Software community** with close connections and collaborations to the experts in NP and HEP (CERN ROOT, Geant4, HEP Software Foundation, Mcnet).

What Simulation Tool(s) To Use?

- Completing the detector proposals on time will require that software efforts proceed pragmatically with maximal efficiency along already established paths.
  - This means two (or three) full simulation frameworks in use.
- Migrating either proto-collaboration fully to either framework now would inefficiently consume effort better expended on simulation studies proper.
  - There will be overlap between proto-collaborations and there will likely remain two (or three) full simulation frameworks.

- What is essential is that the subsystem simulations come together in an integrated way to a complete coherent detector model.
- The SWG has provided a software infrastructure that can support this pragmatic, efficient approach:
  - Geant4 detector subsystem models are modular enough that they can be integrated in either ESCalate or Fun4All (cf. tutorials in early 2020, slide 9) or by extension in a next-generation common framework.
  - EicToyModel exists as a starting point for detector integration across subdetector models.

Towards a Next-Generation Common Framework

ELECTRON ION COLLIDER USER GROUP
STATE OF SOFTWARE SURVEY

Survey from February 16 – 23, 2021. **report**

There are **too many generators and simulation tools** used at the moment.

Unify the Simulation Effort

- The SWG is preparing to launch a **common effort on next-generation simulations**:
  - building on the work done in the existing simulations,
  - unify the software community behind one common effort,
  - a requirement for the common framework is that it integrate the existing detector simulations in a modular way.

Makoto Asai (SLAC)

Our Vision Forward

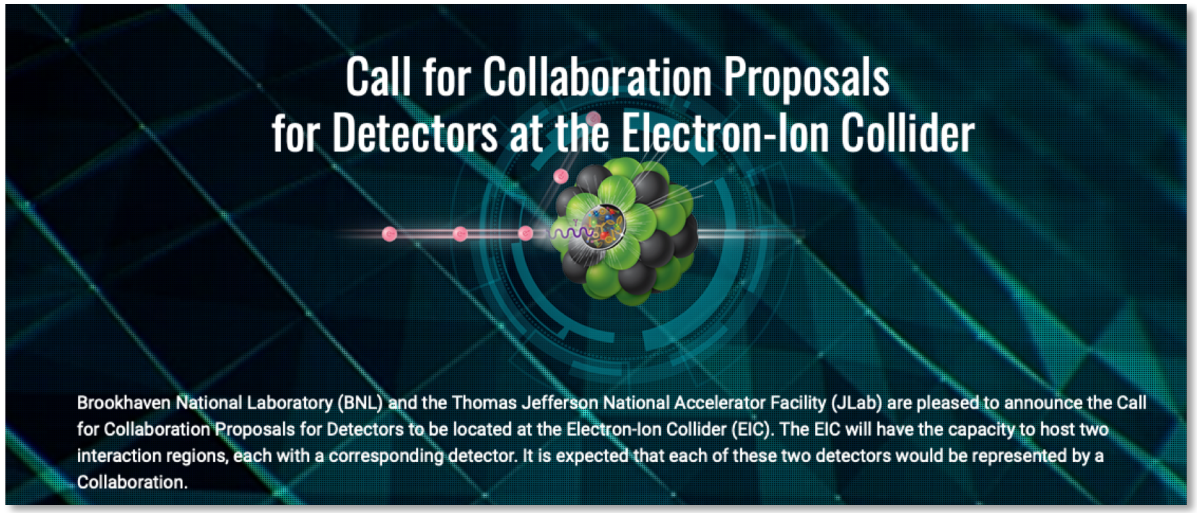
Unify the Simulation Effort with the Community

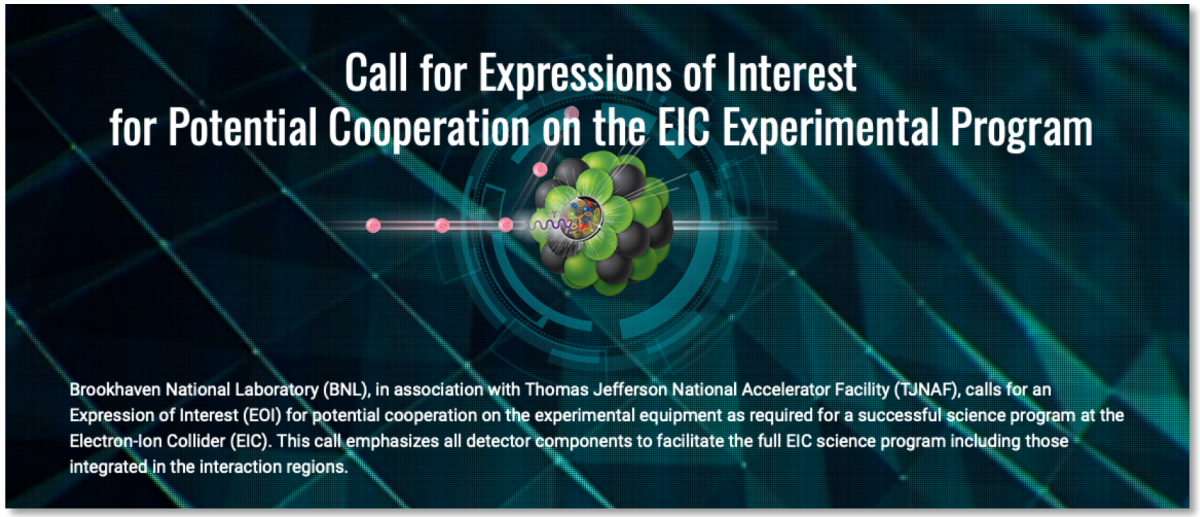
- The SWG is preparing to launch a **common effort on next-generation simulations**:
  - Fast and full simulations in Geant4 for large-scale detector systems with a plug and play modular approach
    - Building on the work done in the existing simulations
    - A requirement for the common framework is that it integrate the existing detector simulations in a modular way.

Community Scenarios

1. If the proto-collaborations proceed in the context of working pragmatically with present tools now but anticipating a migration to a common framework after the current phase, in 2022, then this year's work across the proto-collaborations can be convergent, with both in their evolution targeting compatibility with one modular common framework, the architecture and APIs of which will take shape this year, in close collaboration with the existing simulation developers.
2. If however the proto-collaborations proceed with the mindset and plan that they have chosen their framework now, then the software stacks of the proto-collaborations will diverge greatly over the next year and likely make a common framework impractical for good.

Summary





Support of Collaboration Proposals

get work done in the short term

Realize Software EoI

sustainable effort

Weekly Meetings https://indico.bnl.gov/category/301/

Mailing List eicug-software@eicug.org

Get involved