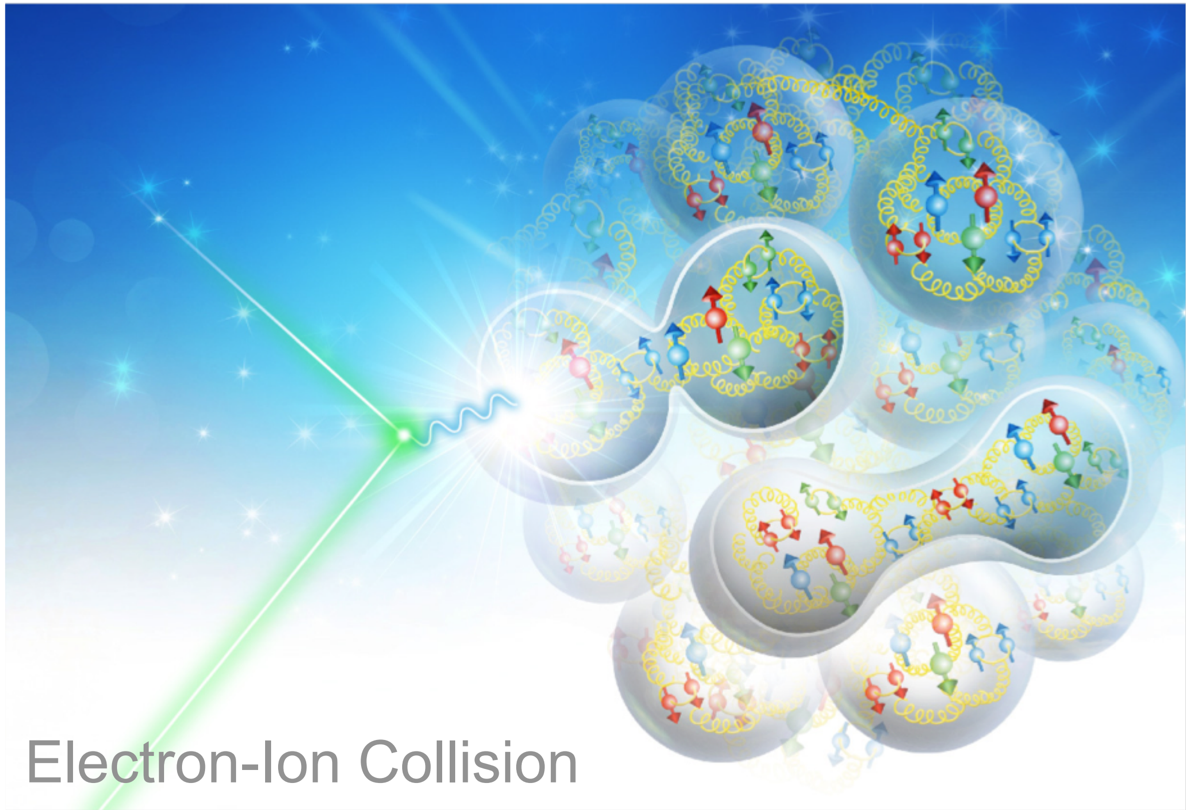
EICUG Software Working Group

Andrea Bressan (INFN, University of Trieste)
Markus Diefenthaler (EIC^2, Jefferson Lab)
Torre Wenaus (Brookhaven Lab)



BROOKHAVEN NATIONAL LABORATORY
Jefferson Lab
UNIVERSITÀ DEGLI STUDI DI TRIESTE

Conveners

Andrea Bressan
(INFN, University of Trieste)

Markus Diefenthaler
EIC^2, Jefferson Lab

Torre Wenaus
Brookhaven Lab

Charge for EICUG Software Working Group

The EICUG Software Working Group's initial focus will be on **simulations of physics processes and detector response** to enable quantitative assessment of measurement capabilities and their physics impact. This will be pursued in a manner that is accessible, consistent, and reproducible to the EICUG as a whole.

It will embody simulations of all processes that make up the EIC science case as articulated in the white paper, eventually integrating new processes under request and with the help of interested communities within the EICUG. The Software working group is to **engage with new major initiatives that aim to further develop the EIC science case**, including for example the upcoming INT program(s), and is anticipated to play key roles also in the preparations for the EIC project(s) and its critical decisions. The working group will build on the considerable progress made within the EIC Software Consortium (eRD20) and other efforts. The evaluation or development of experiment-specific technologies, e.g. mass storage, clusters or other, are outside the initial scope of this working group until the actual experiment collaborations are formed.

The working group will be open to all members of the EICUG to work on EICUG related software tasks. It will communicate via a new **mailing list** and organize **regular online and in-person meetings** that enable broad and active participation from within the EICUG as a whole.

User requests

Ongoing EIC project
Software ✓
Documentation ✓
Requests none

EIC User Group
Common Software ✗
Common Documentation ✗
Requests software, documentation

EIC Generic Detector R&D projects
Software ✓
Documentation ✗ - ✓
Requests common software

Example projects
- ANL: TOPSiDE LDRD
- BNL: eRHIC preCDR
- JLAB: JLEIC preCDR

Focus on (pre)**CDRs and site selection as part of CD1** using existing lab software

Focus on preparation of EIC collaborations
- further develop EIC Science
- examine detector requirements
- work on detector designs
- work on detector concepts

requires simulations of physics processes and detector response

Request from Thomas Ullrich, manager of the R&D program:
- in most cases only GEANT simulations are needed:
  - no need for sophisticated framework
  - no need for elaborate tracking
- a simple *lite setup* with a well defined geometry description standard might get them a long way as long if it is EIC wide and easy to use

EIC Software Groups (beyond the simulation effort at the labs)

High Energy Physics

CERN ROOT
Possible collaboration

HEP Software Foundation
Started collaboration

MCnet
Started collaboration

SLAC Geant4
Established collaboration

Nuclear Physics

EIC Software

EIC Software Consortium
Community Endorsement ✗
Funding ✓ (EIC Generic Detector R&D)

EICUG Software Working Group
Community Endorsement ✓
Funding ✗

Same software suite Seamless data processing from DAQ to data analysis using AI

EIC Streaming Readout Consortium
Community Endorsement ✗
Funding ✓ (EIC Generic Detector R&D)

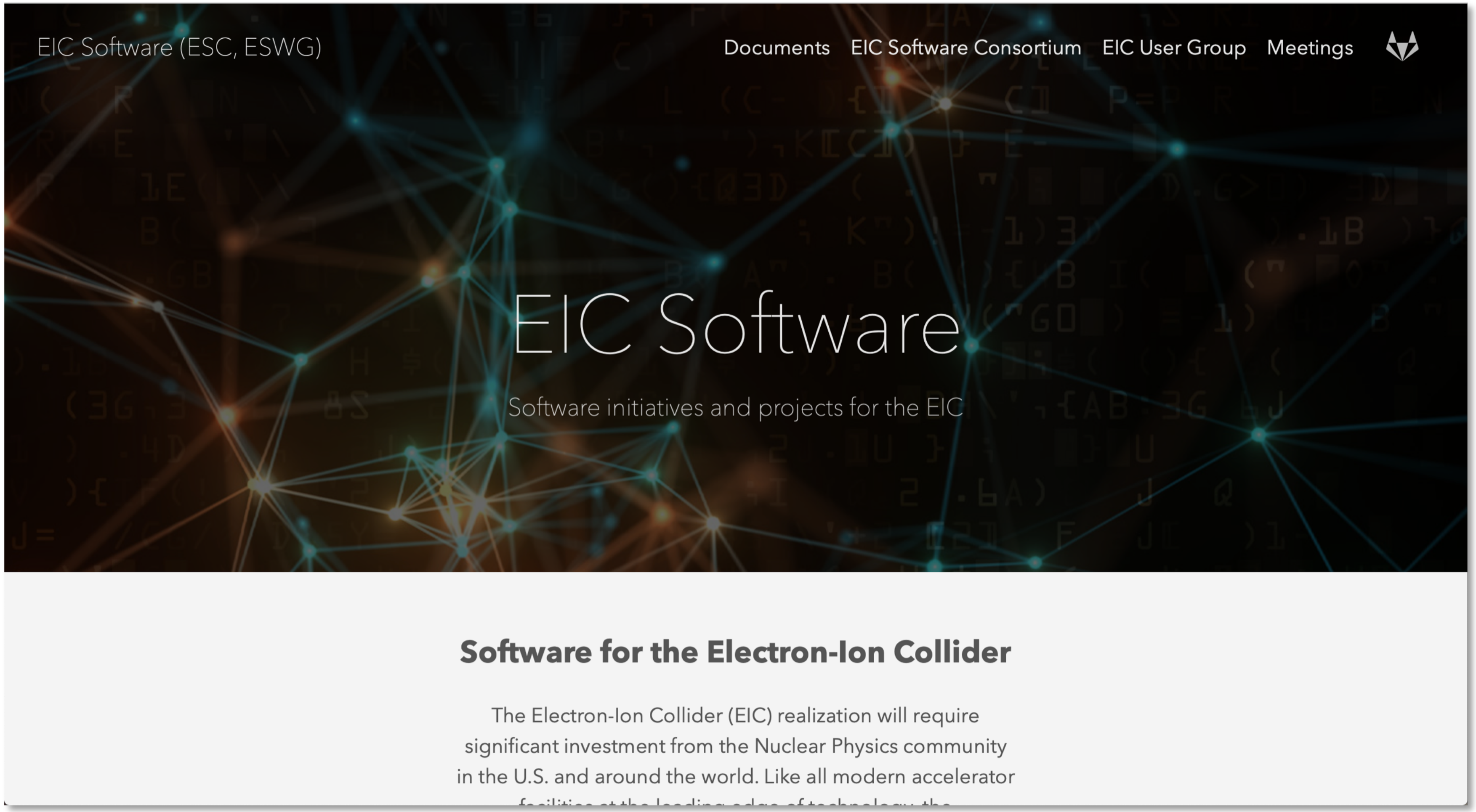
EIC Software for wider community

Workflow environment for EICUG

to use (tools, documentation, support) **and**

to grow with user input (direction, documentation, tools)

Single point of entry



EIC Software website
URL https://eic.gitlab.io

Introduction
- EIC Software Consortium
- EICUG Software Working Group

Meeting schedule

Documents
- container guidelines
- quick start tutorial

Repository

JupyterLab environment

- **collaborative workspace** to create and share Jupyter Notebooks
- **web-based interactive analysis environment** accessible, consistent, reproducible analyses
- **fully extensible and modular** build a collection of analyses and analysis tools
- **bridge to modern data science**, e.g.,
  - *Nature* **563**, 145-146 (2018): “Why Jupyter is data scientists’ computational notebook of choice”
  - more than three million Jupyter Notebooks publicly available on GitHub

07/23 EIC Software Tutorial
Dmitry Romanov (JLAB) introduced EIC simulations in JupyterLab environment.
Quickstart https://eic.gitlab.io/documents/quickstart/

Jupyter Notebooks

- **writing analysis code**

```
[4]: jana.plugin('hepmc_reader') \
     .plugin('jana', nevents=10000, output='hepmc_sm.root') \
     .plugin('eic_smear', detector='jleic') \
     .plugin('open_charm')
```

[4]: **eJana** configured
plugins: hepmc_reader,eic_smear,open_charm

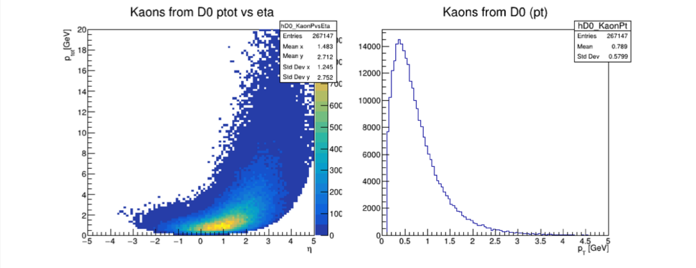
```
[5]: jana.source('../data/herwig6_20k.hepmc')
```

[5]: **eJana** configured
plugins: hepmc_reader,eic_smear,open_charm
sources:
../data/herwig6_20k.hepmc

```
[6]: jana.run()
```

Total events processed: 10001 (~ 10.0 kevt)

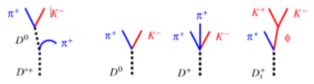
- **visualization of results**



- **narrative of the analysis**

Open charm

The high luminosity at the EIC would allow measurements of open charm production with much higher rates than at HERA and COMPASS, extending the kinematic coverage to large $x_B >\sim 0.1$ and rare processes such as high-p_T jets. Heavy quark production with electromagnetic probes could for the first time be measured on nuclear targets and used to study the gluonic structure of nuclei and the propagation of heavy quarks through cold nuclear matter with full control of the initial state.



Software design

Escaping complexity scaling trap

- provide interfaces to internal layers
- interaction between layers must be clear

Modularity each layer must be replaceable

simple	JupyterLab web interface
moderate	analysis scripts, python
complex	eJANA, plugins, C++
expert	JANA, eic-smear, ROOT, Geant4

Possible JupyterLab environment for EicROOT, fun4all, etc.

Simulations of physics processes and detector responses

Simulation of physics processes

Simulation of detector responses

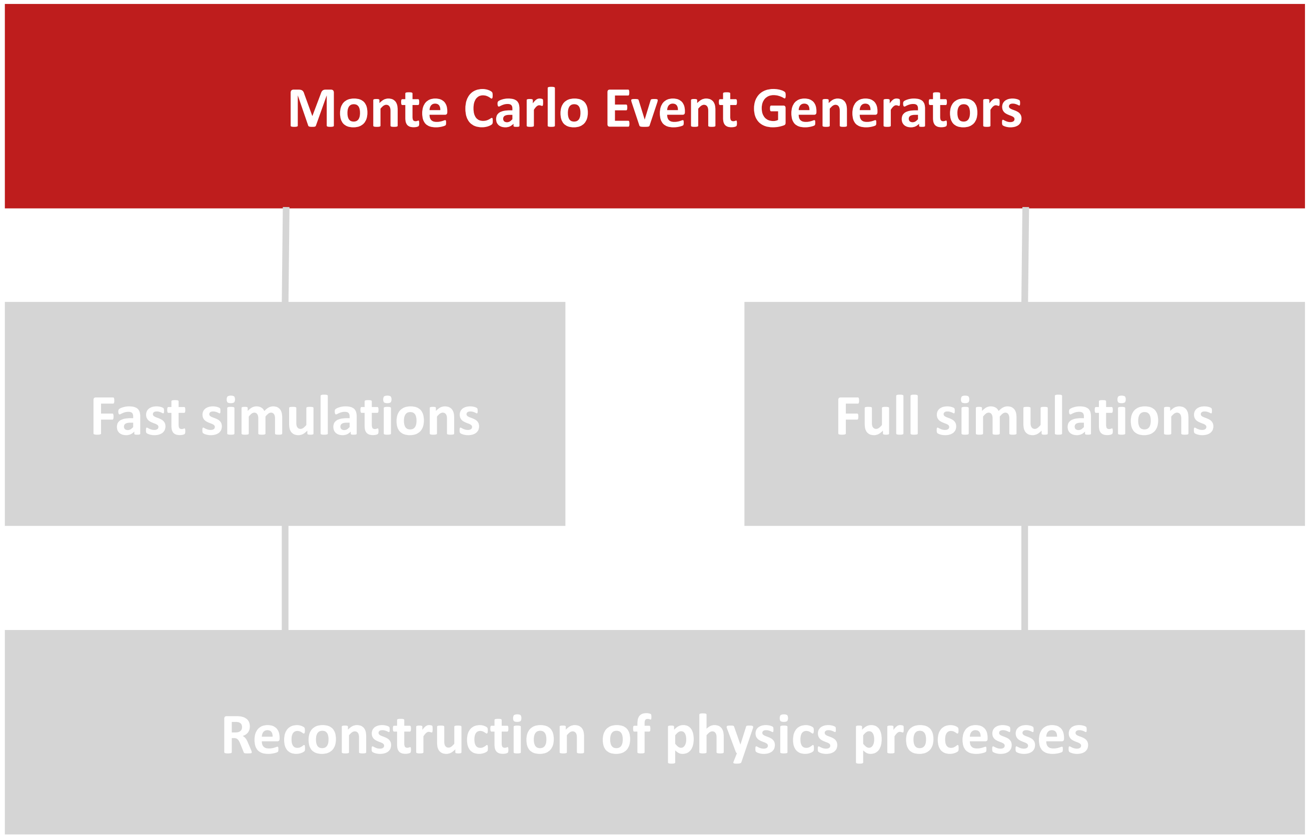
Physics analysis

Simulations of physics processes and detector responses

Simulation of physics processes

Simulation of detector responses

Physics analysis



Workshops: MCEGs for future ep and eA facilities

MCnet 16 institutions, 80+ scientists

Workshop series

- EICUG—Mcnet collaboration
- started as satellite workshop during POETIC-8

Goals

- requirements for MCEGs for ep and eA
- R&D for MCEGs for ep and eA

MCEG2019: Next steps

- **General-purpose MCEGs**
  - HERWIG, PYTHIA, and SHERPA, will be significantly improved w.r.t. MCEGs at HERA time.
  - MCEG-data comparisons will be critical to tune the MCEGs to DIS data and theory predictions.
  - The existing general-purpose MCEG should soon be able to simulate NC and CC unpolarized observables also for eA. A precise treatment of the nucleus and its breakup is needed.
  - First parton showers and hadronization models for ep with spin effects, but far more work needed for polarized ep / eA simulations.
  - Need to clarify the details about merging QED+QCD effects (in particular for eA).
- **TMD physics**
  - Vibrant community working on various computational tools for TMDs.
  - CASCADE: MCEG for unpolarized TMDs at high energy.
  - Need more verification of MCEG models with TMD theory / phenomenology.
- **GPD physics**
  - No modern MCEG available.
  - There is a path from PARTONS to a GPD MCEG, similar there is a project to extend MCEG for exclusive processes from JLAB12 to EIC.

See Ilkka Helenius *"General-purpose Monte Carlo event generators for an EIC"* (07/25)

MCEG–HERA comparisons and MCEG validation for ep

Rivet example

SIDIS analysis at HERMES

```
// Extract the particles other than the lepton
const FinalState& fs = apply<FinalState>(event, "FS");
Particles particles;
particles.reserve(fs.particles().size());
const GenParticle* dislepGP = dl.out().genParticle();
foreach (const Particle& p, fs.particles()) {
  const GenParticle* loopGP = p.genParticle();
  if (loopGP == dislepGP)
    continue;
  particles.push_back(p);
}

// Apply HERMES cuts.
bool validx = (x > 0.023 && x < 0.6);
if (q2 < 1. || w2 < 10. || y < 0.1 || y > 0.85 || !validx)
  vetoEvent;

// good inclusive event, let's do bookkeeping before we look at the hadrons
dis_tot += weight;
dis_x->fill(x, weight);
dis_Q2->fill(q2, weight);

for (size_t ip1 = 0; ip1 < particles.size(); ++ip1) {
  const Particle& p = particles[ip1];

  // get the particle index, check if it is a particle of interest
  const int part_idx = get_index(p.genParticle()->pdg_id());
  if (part_idx < 0) {
    continue;
  }

  // we have a particle of interest, let's calculate the kinematics
  // z
  const double z = (p.momentum() * pProton) / (pProton * q);
  // pt
  const double pth = sqrt(p.momentum().pT2());

  // get our z index, if negative, we have a particle outside of [.2, .8]
  const int z_idx = calc_zslice(z);
  if (z_idx < 0) {
    continue;
  }

  // store the events and make cuts where necessary
  //
  // pt cut for variables not binned in pt
  if (pth > 0 && pth < 1.2) {
    mult_z[part_idx]->fill(z, weight);
    mult_zx[part_idx][z_idx]->fill(x, weight);
    mult_zQ2[part_idx][z_idx]->fill(q2, weight);
  }
  mult_zpt[part_idx][z_idx]->fill(pth, weight);
}
```

- **MCEG R&D** requires *easy* access to *data*
- data := analysis description + data points

- **HEP** existing workflow for MCEG R&D using tools such as Rivet and Professor

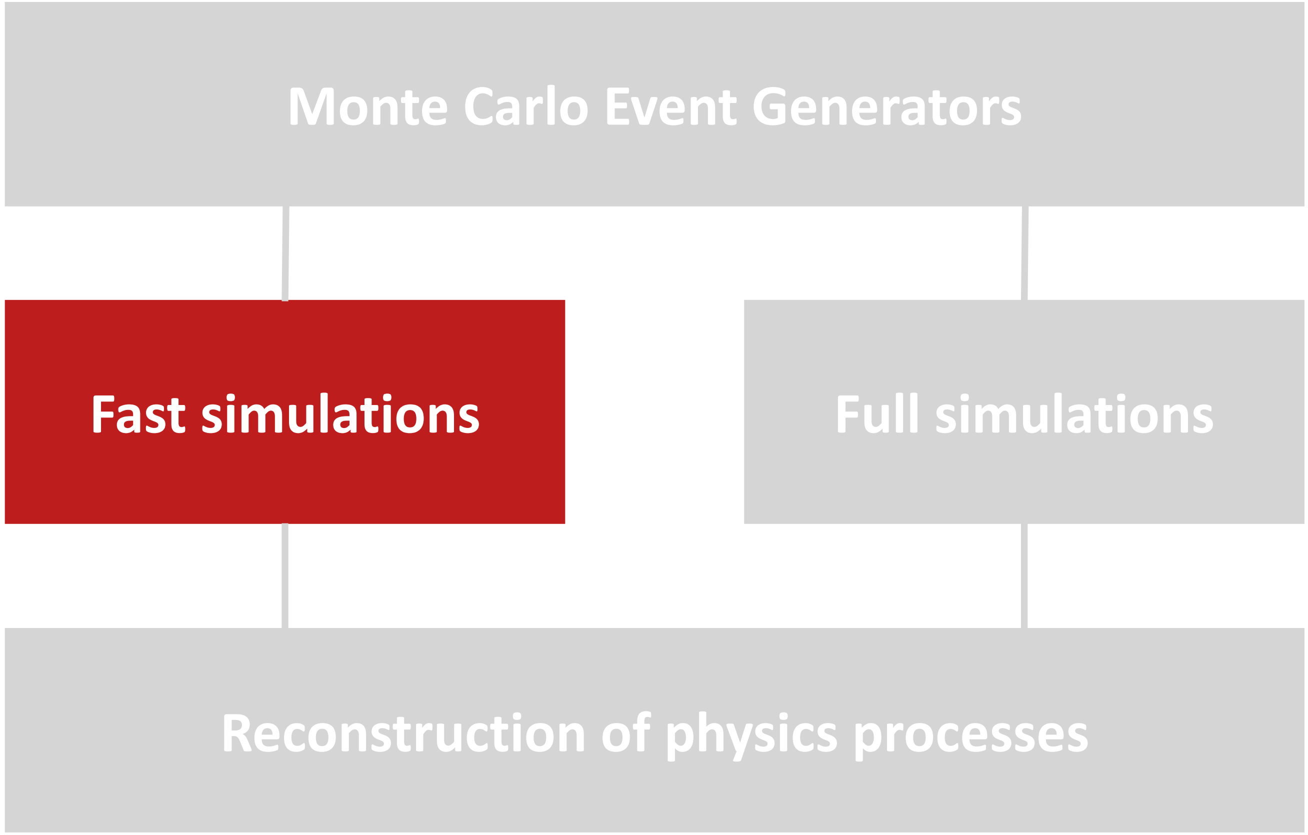
- **Detailed comparisons between modern MCEG and HERA data**
  - workshop on Rivet for ep (Feb 18—20 2019)
  - mailing list rivet-ep-l@lists.bnl.gov
  - HERA data not (yet) included in MCEG tunes

Simulations of physics processes and detector responses

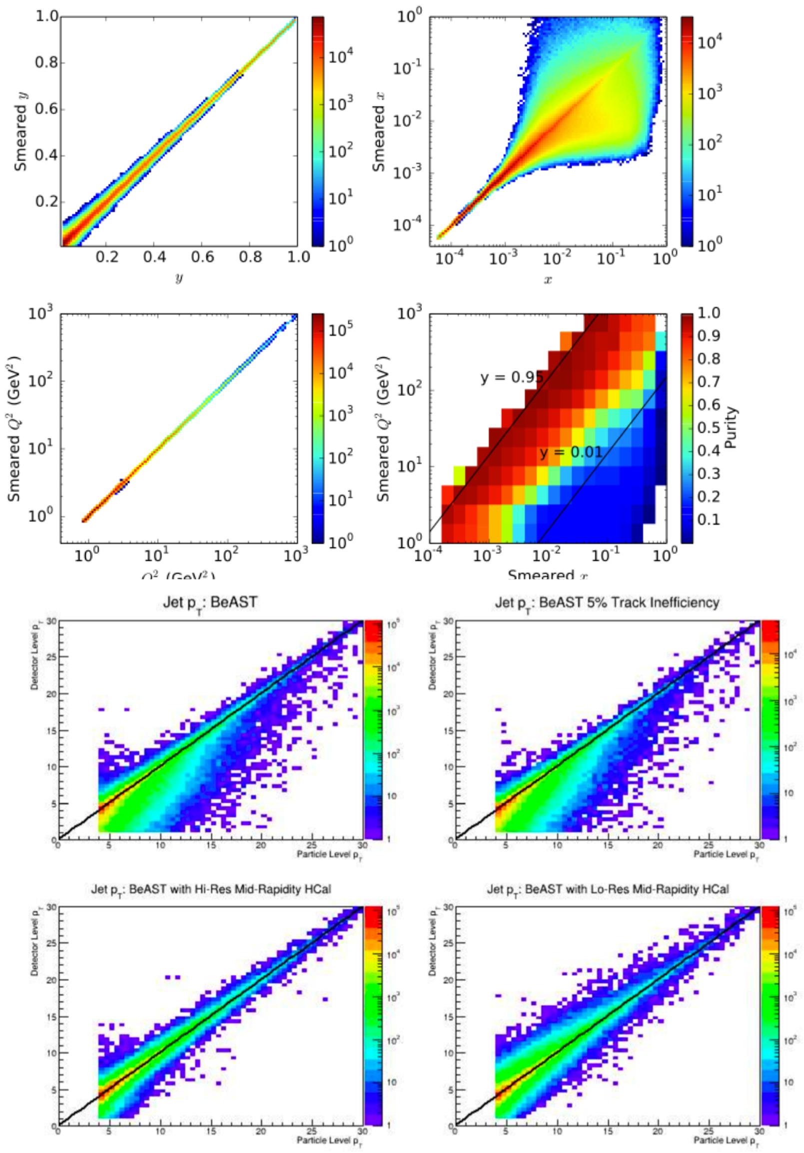
Simulation of physics processes

Simulation of detector responses

Physics analysis



K. Kauder (BNL), A. Kiselev (BNL)



Fast simulations using ROOT, ideal for questions like

- *"Given a (known) detector performance, how well can I measure some physics observable(s)?"*
- *"If I need to measure X to some precision, what detector performance do I need?"*
- Used extensively for **EIC White Paper**

Features

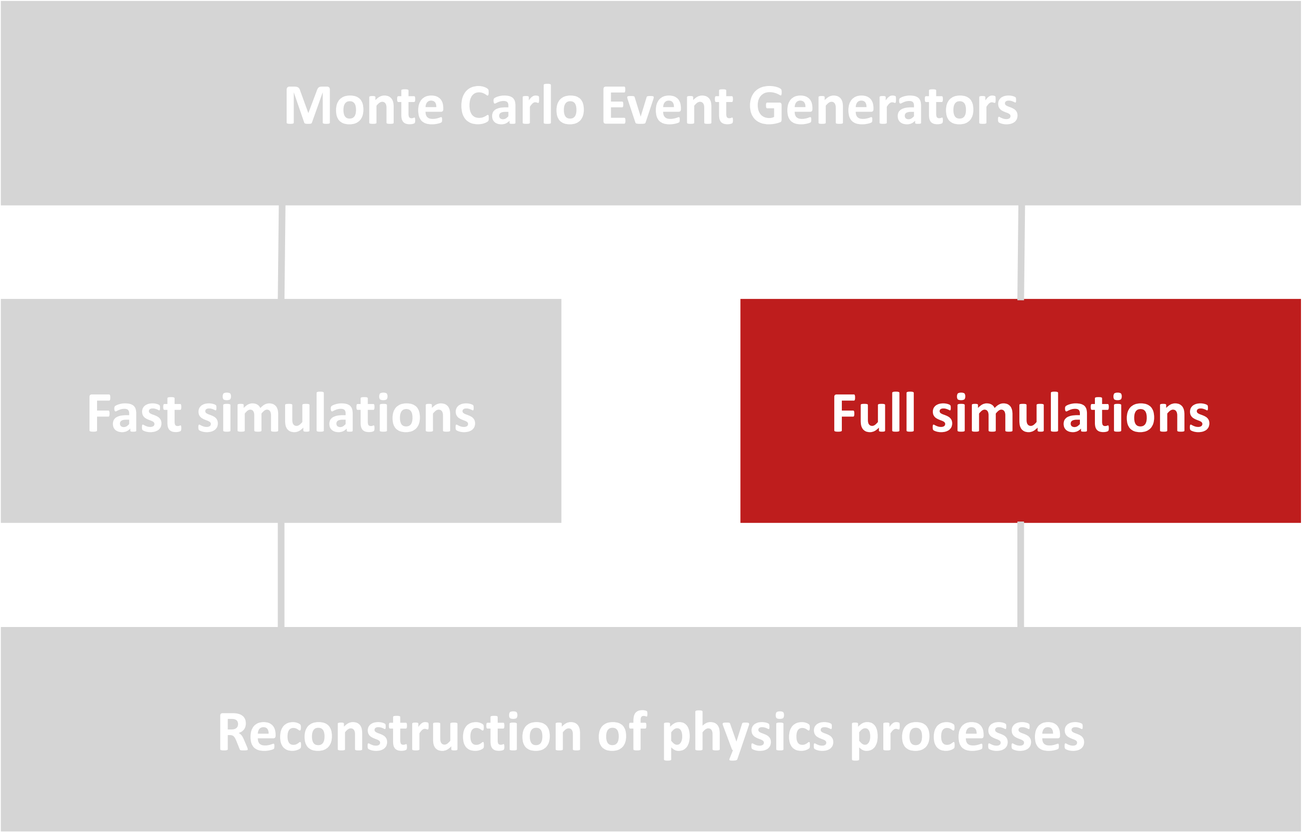
- interface to MCEGs for ep and eA smearing of overall detector performance:
  - can be easily modified in user code
  - includes acceptance effects
  - parametrizations for BeAST, ePHENIX, JLEIC and others
- ROOT trees for MC Truth and smeared information

Simulations of physics processes and detector responses

Simulation of physics processes

Simulation of detector responses

Physics analysis



Examples for existing EIC Software

ANL TOPSiDE detector concept (ILC software variant)

BNL BeAST detector concept: EICroot (FairRoot variant)

BNL ePHENIX detector concept (Fun4All)

JLAB JLEIC detector concept (GEMC → eJANA)

Used for various EIC detector and physics studies including NAS reports, white papers, pre-conceptual design reports, etc.

Full detector simulation in Fun4All

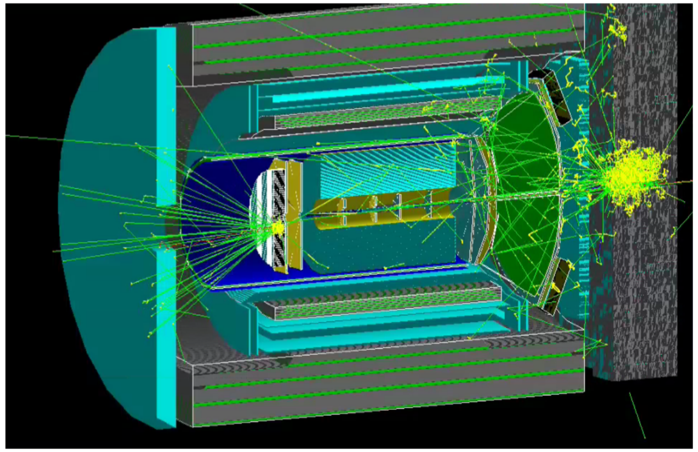
C. Pinkenburg (BNL)

Fun4All ROOT sample for configuring and running detector simulations in Geant4

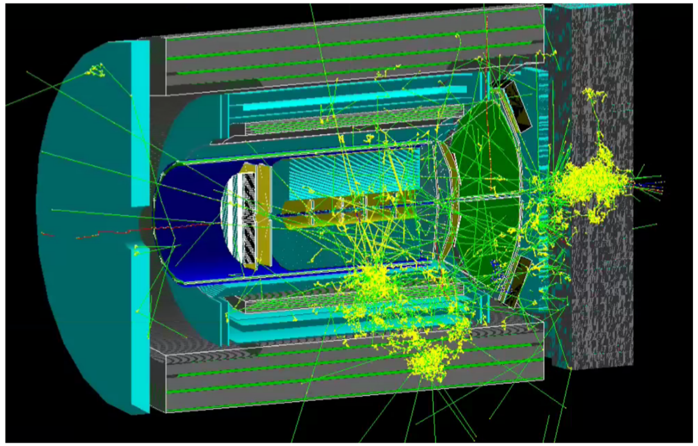
```
// EICDetector geometry - barrel
bool do_DIRC = true;

// EICDetector geometry - 'hadron' direction
bool do_RICH = true;
bool do_Aerogel = true;

bool do_FEMC = true;
bool do_FEMC_cell = do_FEMC && true;
bool do_FEMC_twr = do_FEMC_cell && true;
bool do_FEMC_cluster = do_FEMC_twr && true;
bool do_FEMC_eval = do_FEMC_cluster && true;

bool do_FHCAL = true;
bool do_FHCAL_cell = do_FHCAL && true;
bool do_FHCAL_twr = do_FHCAL_cell && true;
bool do_FHCAL_cluster = do_FHCAL_twr && true;
bool do_FHCAL_eval = do_FHCAL_cluster && true;

// EICDetector geometry - 'electron' direction
bool do_EEMC = true;
bool do_EEMC_cell = do_EEMC && true;
bool do_EEMC_twr = do_EEMC_cell && true;
bool do_EEMC_cluster = do_EEMC_twr && true;
bool do_EEMC_eval = do_EEMC_cluster && true;
```

- One EIC Detector fully implemented (ePHENIX)
- Subdetectors are easy to switch on and off for detector studies
- Event Generators available via on/off switches
- Optional overlap check can be run for debugging new additions
- Full reconstruction implemented for calorimeters and tracking
- Jet reconstruction also available via on/off switch

Pythia6: DIS (e (18GeV), p (275 GeV)) Fun4All_G4_EICDetector.C



Neutral Current DIS



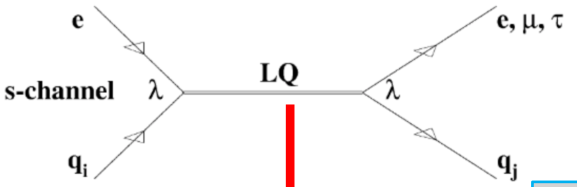
Charged Current DIS

Courtesy of Dillon Fitzgerald, Enrique Gamez, Kara Mattioli, Cynthia Nunez, Desmond Shangase from University of Michigan

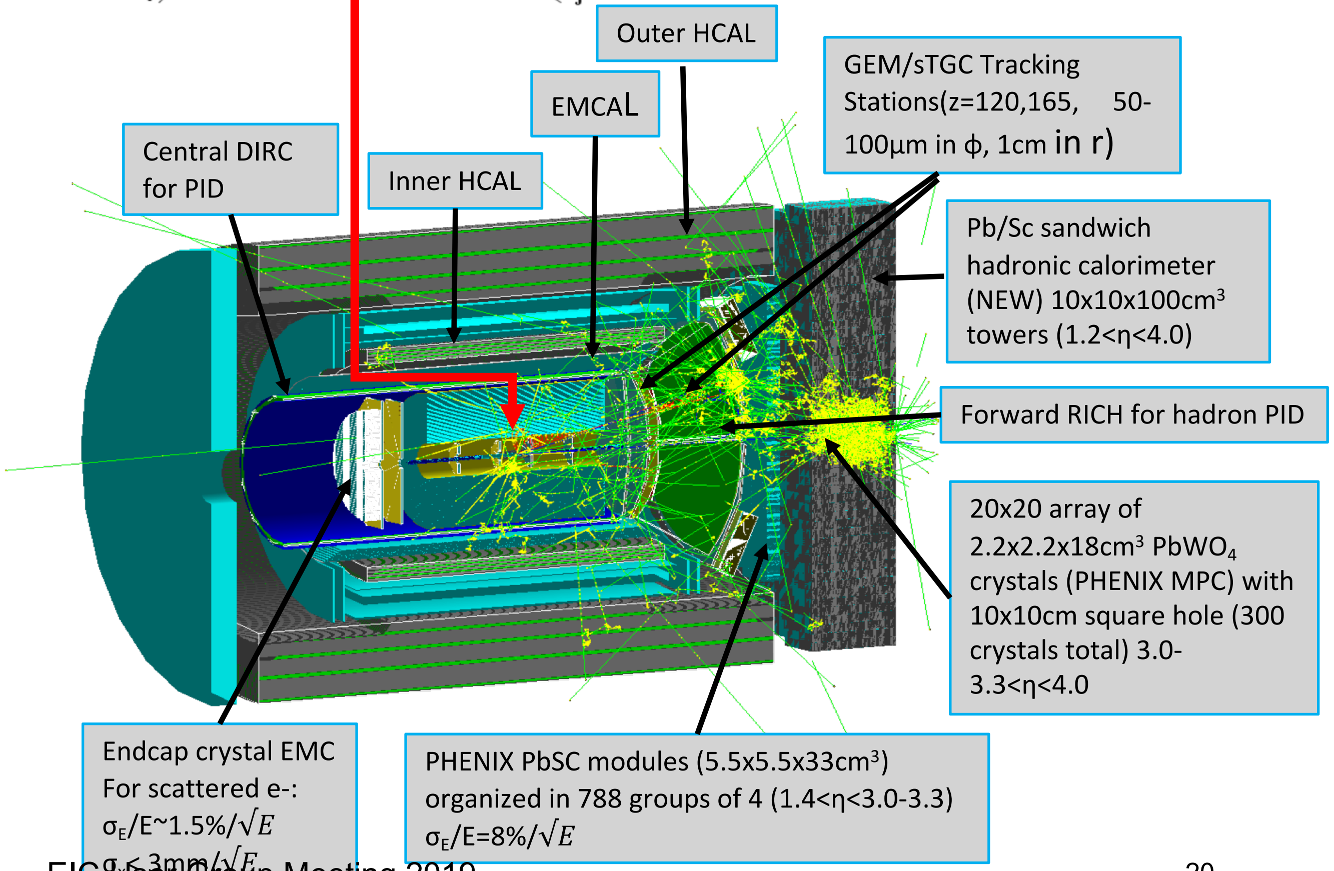
Example ePHENIX physics study

C. Pinkenburg (BNL)

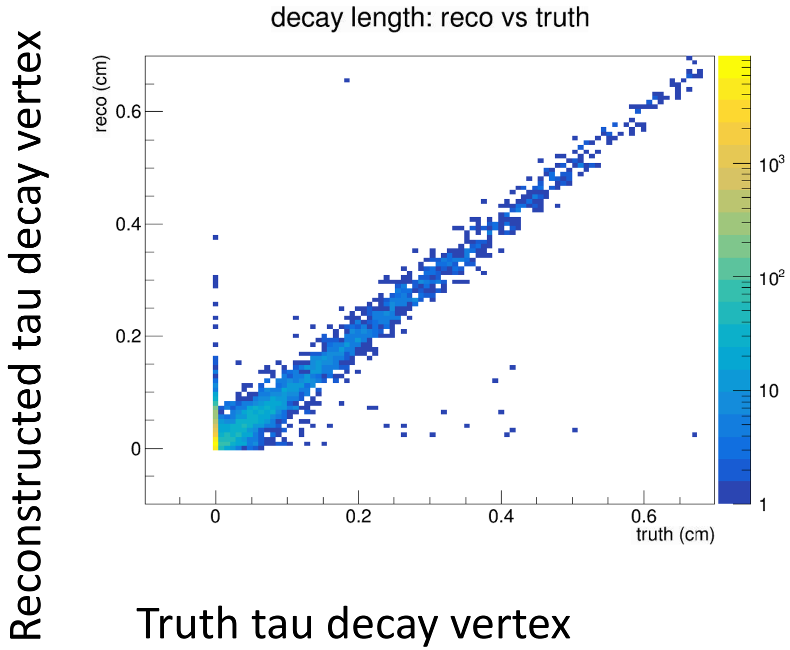
- Full detector **simulation, tracking, vertex** finding using Fun4All infrastructure



- LQGENEP for signal
- DJANGOH/PYTHIA for SM bkgd



07/23 Parallel talk Searching for τ-lepton appearance at the EIC at the highest luminosities (Krishna Kumar et al.)



Courtesy of Jinlong Zhang (SBU)

Detector Simulation

- **collaboration with Geant4 International Collaboration**
  - liaison: Makoto Asai (SLAC)
- **Detector Simulation R&D**
  - containers and tutorials for EIC detector simulations
  - coordinate input for Geant4 validation based on EIC physics list maintained by SLAC Geant4 group

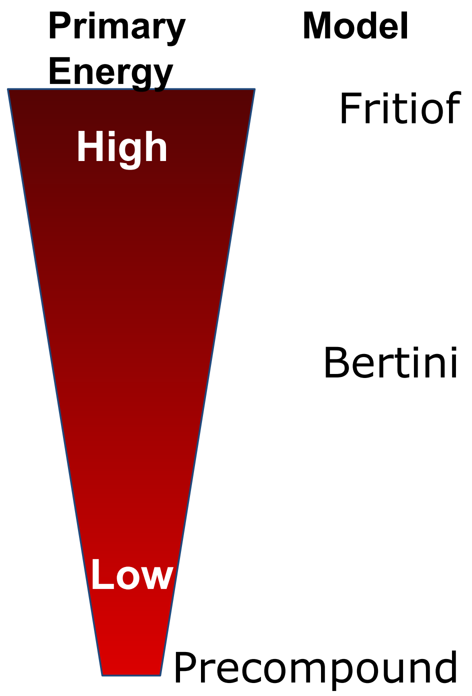
09/23 – 09/27 Geant4 Collaboration Meeting at Jefferson Lab

09/24 Geant4 Technical Forum on EIC

- Geant4 developers will update the EIC community of the recent and ongoing developments that are relevant for EIC
- requirements and concerns from the EIC community should be presented and discussed
- Makoto Asai and MD are coordinating input

EIC

- energy range is different from LHC
- validation, tuning and extension including test beam studies



Addressing user requests (slide 4)

EIC Software Meeting on Detector and Physics Simulations

Wednesday 10 Jul 2019, 09:00 → 15:00 US/Eastern

2-160 (BNL Physics)

Andrea Bressan (Trieste), Markus Diefenthaler (Jefferson Lab), Torre Wenaus (BNL)

35 participants

Why we urgently need a
- common (EIC-wide)
- easy to use*
- capable

detector simulation software

EIC Software Meeting on Detector and Physics Simulations

Wednesday Jul 10, 2019, BNL

Thomas Ullrich (BNL)

* for a grad-student/postdoc with moderately good computing skills

EIC User Group
- preparation of EIC collaborations
- request for common software tools and documentation

Request from EIC Generic Detector R&D program:
- in most cases only GEANT simulations are needed:
  - no need for sophisticated framework
  - no need for elaborate tracking
- a simple *lite setup* with a well defined geometry description standard might get them a long way as long if it is EIC wide and easy to use

Common Geant4 infrastructure

Goals

- meet requirements by EIC community fully
- meet requirements by EIC community by end of 2019

Approach

- common repository for detector R&D for tEIC
- common detector description in Geant4 (C++)
- common detector naming convention for EIC
- common definition of parameters and their management
- common API/class design for sensitive detector stepping action
- possible common hits output structure
- concise document and template on how to implement and integrate subdetector in the detector concepts for the EIC

Common Geant4 infrastructure compatible with existing simulations tools, e.g., eJANA or fun4all.

Existing tools will be able to **use common Geant4 infrastructure** without loosing any functionality.

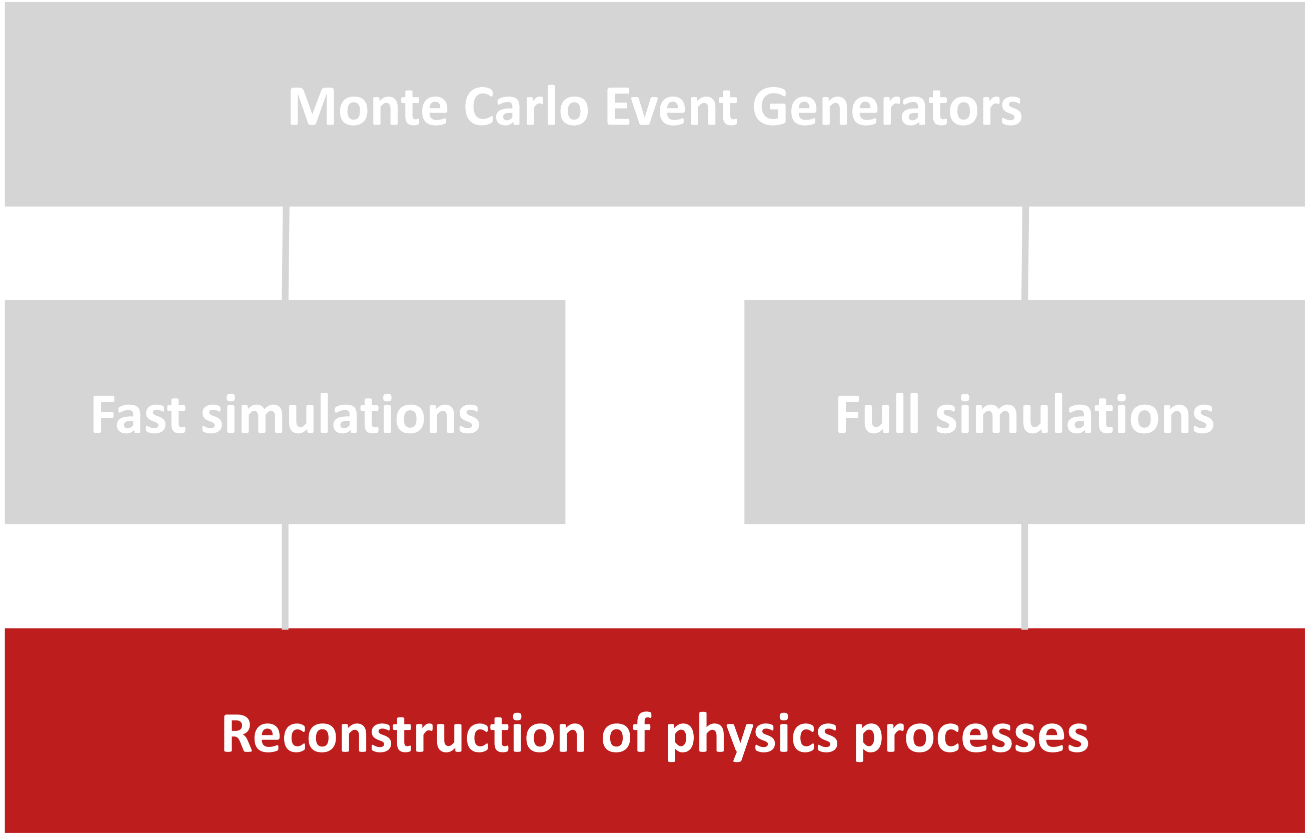
Existing prototypes for common Geant4 infrastructure: EIC Software Sandbox, (Alexander Kiselev), g4e (Yulia Furletova et al.)

Simulations of physics processes and detector responses

Simulation of physics processes

Simulation of detector responses

Physics analysis



e[JANA] - stands for EIC JANA

- Basic reconstruction
- Physics analysis
- **Users detector codebase integration**

Any existing C++ (or even others) code can be:
- compiled as JANA plugin
- run parallelized in eJANA
- accessed by other plugins

Reconstruction

- **Tracking** - Genfit
- **Vertex finding** – Rave
- **Physical analysis**:
  - ROOT C++ or
  - Python data science tools (Jupyter, Seaborn, Pandas, etc.)

Working for EIC Community

Engage with EIC Software initiatives

Building an EIC Software community

Engage with EIC Software Working Group

Google Group

- join using Google account: sign up directly
- Join non-using Google account: contact us and we will add you asap

Mailing list eicug-software@eicug.org

- subscribe via Google Group

EICUG Software Working Group

- https://gitlab.com/eic

EIC Software Consortium

- https://gitlab.com/ESC

Join GitLab repositories with your GitLab account

Meeting schedule in 2019

Date	Topic(s)
02/14	Documentation, tutorials, and website (remote)
02/20 – 02/22	MCEG for future ep and eA facilities (DESY Hamburg, Germany)
03/07	MCEG2019 report, MC for EIC (remote)
04/11	Making EIC simulations more accessible for users (remote)
05/20 – 05/22	EIC Software Meeting (Trieste, Italy)
06/27	Benchmarks and validation (remote)
07/10	EIC Ad-hoc Meeting on detector and physics simulations (BNL)
07/23	Tutorial during EICUG Meeting (Paris, France)
08/	Summer break
09/24	Geant4 Technical Forum on EIC

TBA Monthly remote meetings, in-person meetings, tutorials

Remote meetings for worldwide attendance

11:00 a.m. (EDT)

7:00 p.m. (EDT)

Andrea Bressan (INFN, University of Trieste)
Markus Diefenthaler (EIC^2, Jefferson Lab)
Torre Wenaus (Brookhaven Lab)
eicug-software@eicug.org

EICUG Software Working Group

Merci beaucoup à David Blyth:

- 06/2018 – 01/2019 convener until leaving to industry

EIC Simulations:

- **JupyterLab** collaborative workspace for EICUG
- develop common tools to meet community requirements: **common Geant4 infrastructure**
- provide **single point of entry** simulations (website with repository, documentation, tutorials)
- **engage with EICUG** monthly remote meetings, in-person meetings, tutorials
- **engage with software community** Geant4, HSF, MCnet, and ROOT

BROOKHAVEN NATIONAL LABORATORY

Jefferson Lab

UNIVERSITÀ DEGLI STUDI DI TRIESTE