

Developing Analysis Tools and Techniques for the EIC

Proponents: Whitney Armstrong (ANL), Elke-Caroline Aschenauer (BNL), Franco Bradamante (INFN Trieste), Andrea Bressan (INFN Trieste), Andrea Dotti (SLAC), Sergei Chekanov (ANL), **Markus Diefenthaler (Jefferson Lab, co-PI)**, **Alexander Kiselev (BNL, co-PI)**, Anna Martin (INFN Trieste), Christopher Pinkenburg (BNL), Stefan Prestel (SLAC)

Contact: Markus Diefenthaler (mdiefent@jlab.org)

Abstract

The EIC realization will require significant investment from the nuclear science community in the US and around the world. Like all modern accelerator facilities at the leading edge of technology, the computational demands will be sizeable. To realize the physics program laid out in the White Paper, the high-luminosity machine needs to be matched by detectors capable of delivering motivating science. The success of detector designs depends on our ability to accurately simulate their response and analyze their physics performance. Therefore, early investment in the development of software tools will have an immense impact on the quality of the future scientific output. With this in mind we propose to identify and develop the required simulation and analysis tools by forming a software consortium. In this proposal we begin with an outline of forward-looking global objectives that we think will help sustain a software community for more than a decade. We then identify the high-priority projects for immediate development in FY17 and also those, which will ensure an open-source development environment for the future.

Overview

History

In the Generic R&D meeting in January 2016, Elke-Caroline Aschenauer, Markus Diefenthaler, and Alexander Kiselev have presented a letter of intent for “Developing Analysis Tools and Techniques for the EIC” [1]. The R&D committee has welcomed our “*initiative and agrees that a robust software environment, compatible with the existing software frameworks, is very important for the development of the physics case for the EIC.*” [2]. They suggested to reevaluate our “*strategy and consider a long-term perspective in the development of the basic infrastructure in terms of geometry definition, i/o interface and analysis tools, that guarantees its long-term value to the community and ultimately becomes the framework used in the experiments at an EIC.*”

Since January 2016, scientists from ANL, BNL, INFN Trieste, Jefferson Lab and SLAC have joined efforts to form a strong collaboration. We have identified specific project goals for FY17 and we will continue to develop and sharpen our strategic program for the EIC software consortium.

[1] <https://wiki.bnl.gov/conferences/images/6/65/LOI-SoftwareConsortium.pdf>

[2] https://wiki.bnl.gov/conferences/images/c/c3/EIC_RnD_Report_Jan_2016.pdf

Objective

The EIC will revolutionize our understanding of the inner structure of nucleons and nuclei. Developing the physics program for the EIC, and designing the detectors needed to realize it, requires a plethora of software tools and multifaceted analysis efforts. Many of these tools have yet to be developed or need to be expanded and tuned for the physics reach of the EIC. Currently, various groups use disparate sets of software tools to achieve the same or similar analysis tasks such as Monte Carlo event generation, detector simulations, track reconstruction, event visualization, and data storage to name a few examples. With a long-range goal of the successful execution of the EIC scientific program in mind, it is clear that early investment in the development of well-defined interfaces for communicating, sharing, and collaborating, will facilitate a timely completion of not just the planning and design of an EIC but ultimate delivery the physics capable with an EIC.

In our consortium, we aim to develop analysis tools and techniques for the EIC, and facilitate communication and collaboration among current and future developers and users. We will help coordinate the EIC software effort, providing organization and guidance to help seed growth of a software community that will exist for well over a decade. While our localized efforts are typically focused on completing specific tasks or developing certain tools, the consortium will focus also on achieving the following forward-looking goals:

1. **Organizational efforts with an emphasis on communication:** We will help with the organization of the software effort for the EIC by providing documentation about the available EIC software and by maintaining a software repository. This function will be eventually taken over by an official EIC software group. We encourage participation in our consortium and will schedule regular meetings to build an active working group and foster collaboration. We will organize an EIC software workshop during this first year to continue the discussions from our previous workshops [3, 4] and to work towards a common software effort.
2. **Planning for the future with forward compatibility:** We will continue the “Future Trends in Nuclear Physics Computing” workshop [4] to discuss new developments and trends in scientific computing. Incorporating new standards and validating our tools on new computing infrastructures are among the main goals of our consortium.
3. **Interfaces and integration:** Given the current stage of the EIC project, it is too early to define the analysis tools of the EIC. However, it is important to connect the existing frameworks / toolkits and to identify the key pieces for a future EIC toolkit. We will work on interfaces between the existing frameworks / toolkits and aim to collaborate with other R&D consortia and projects in general to integrate their tools into existing frameworks / toolkits. By doing so, we will start to define the key pieces of the EIC toolkit and identify the high-priority R&D projects.

[3] <https://www.jlab.org/conferences/trends2016/>

Plan for FY17

MC development: In the “EIC Software Meeting” in September 2015 [4], we have reviewed the MCEGs that are available for the EIC and identified MCEGs and other Monte Carlo tools that need to be developed. For the development of the EIC analysis, a MCEG for TMDs is urgently required. Elke-Caroline Aschenauer, Markus Diefenthaler, and Stefan Prestel have started to work towards a TMD MCEG. There is a separate project for this task. Thus, we have not included this important project in our funding request for FY17. There is also work required on MCEG for eA processes. The development of DPMJetHybrid, a tool to refine detector requirements for eA collisions in the nuclear shadowing / saturation regime is funded by the Generic R&D program for the EIC (eRD17). Elke-Caroline Aschenauer is part of this project. Within our R&D consortium, we will make fundamental contributions to the development of the following Monte Carlo tools:

- Start the development of a library for simulating radiative effects (page 5)
- Validation of critical Geant4 physics in the energy regime of the EIC (page 10)
- Start the development of an universal event display for MC events (page 12)
- Promote open-data developments for efficient data-MC comparisons (page 13)

In the first year, we will also work on these tasks:

- Work towards a common geometry and detector interface (page 15)
- Work towards an unified track reconstruction (page 17)
- Develop interfaces to forward compatible, self-descriptive file formats (page 19)
- Build a community website and organize software repositories dedicated to the EIC

In the following pages, each task is described in detail. The work on each of task listed above is coordinated separately. In regular meetings, we will review the status of each project and discuss the future direction of our work. In the section for each project, we will list which proponent will coordinate the work on this task.

[4] <https://www.jlab.org/conferences/eicsw/>

Funding request for FY17

A focused effort is essential for any R&D effort. We request a travel budget of USD 30,000 to allow proponents to meet and to work together on key tasks or to invite visiting scientists that are essential to the R&D effort. Part of the money will be used for the organization of a workshop and to support the proponent's travel to the workshop.

We request USD 20,000 to fund undergraduate projects for summer students. The undergraduate students will work at ANL, BNL, Jefferson Lab, or SLAC on specific software or analysis tasks.

In total, we request a USD 50,000 for FY17.

Projects for FY17

Consistent approach to integrate radiative corrections into MCEGs (Elke-Caroline Aschenauer)

Goal:

Develop a radiative-correction library for both polarized and unpolarized observables. The library should allow to integrate in MCEGs all radiative effects due to obtain in an unfolding procedure Born level quantities which are needed by theorist to interpret the data.

Detailed Description:

The radiation of real and virtual photons leads to large additional contributions to the observable cross section of electron scattering at high energies. Precision measurements of the nucleon structure require a good understanding of these radiative corrections. For neutral-current lepton nucleon scattering, a gauge-invariant classification into leptonic, hadronic and interference contributions can be obtained from Feynman diagrams. The Feynman diagrams for leptonic corrections are shown in Figure 0-2. Leptonic corrections dominate and strongly affect the experimental determination of kinematic variables.

Usually, inclusive cross sections are measured as a function of Q^2 and Bjorken- x , x_B , defined as

$$Q^2 = -(l-l')^2, \quad x_B = \frac{Q^2}{2P \cdot (l-l')}, \quad \text{where } l \text{ and } l' \text{ denote the 4-momenta of the incoming and outgoing lepton, respectively, and } P \text{ is the 4-momentum of the incoming nucleon. The true values of these variables seen by the nucleon when a photon with 4-momentum } k \text{ is radiated are, however, given by (see$$

Figure 0-1)

$$\begin{aligned} \bar{Q}^2 &= -(l-l'-k)^2, \\ \bar{x}_B &= \frac{\bar{Q}^2}{2P \cdot (l-l'-k)^2} \end{aligned} \quad (1)$$

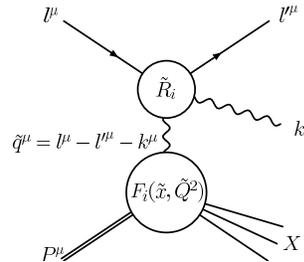


Figure 0-1: Kinematics of leptonic radiation.

If the photon momentum is large and balancing the transverse momentum of the scattered lepton, \bar{Q}^2 can be shifted to small values, leading to an enhancement of the radiative corrections. This effect is similar to the radiative tail of a resonance.

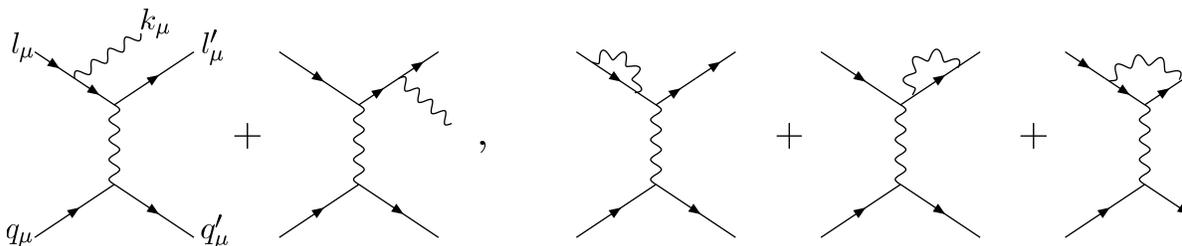


Figure 0-2: Feynman diagrams for leptonic radiation in lepton-quark scattering.

The effect of radiation of photons from the lepton can be described with the help of radiator functions $\bar{R}_i(l, l', k)$. There is one \bar{R}_i for every structure function F_i , $i = 2, L$. The radiator functions comprise both real radiation from the initial and the final state as well as the contribution from vertex and self-energy diagrams. Using \bar{x}_B and \bar{Q}^2 from equation (1) to parameterize the integration over the phase space of emitted photons, one can express the observed structure functions as convolutions, $F_i^{obs}(x_B, Q^2) = \int d\bar{x}_B d\bar{Q}^2 R_i(x_B, Q^2, \bar{x}_B, \bar{Q}^2) F_i^{true}(\bar{x}_B, \bar{Q}^2)$. (2)

The integration limits are determined by the energy allowed for the radiated photon, which in the photon-nucleon center-of-mass frame, is given by $E_\gamma^{\max} = \sqrt{\frac{1-x_B}{x_B} Q^2}$

(3).

Radiative corrections are, therefore, large at large Q^2 and small x_B . In contrast, at small Q^2 and large x_B , the phase space for photon emission is restricted and negative virtual corrections dominate. From equation (2) it is obvious that the determination of the true structure functions $F_i^{true}(\bar{x}_B, \bar{Q}^2)$ requires unfolding, a procedure, which is in general only possible in an iterative way and with reasonably chosen assumptions about the starting values. Moreover, the observed structure functions depend on the way in which the kinematic variables are measured. For example, if the momentum of the hadronic final state, p_X , could be measured, \bar{x}_B and \bar{Q}^2 would be known. In practice this will be difficult to achieve; however, any information about the hadronic final state could contribute to a narrowing down of the phase space available for photon emission, thereby reducing the size of radiative corrections.

The radiator functions are dominated by peaks in the angular distribution for the collinear radiation of photons from the initial state (ISR) or from the final state (FSR). At high energies, it is a good approximation to assume that photon radiation can be described by a simple rescaling of the lepton momentum, $l \rightarrow zl$ for ISR and $l' \rightarrow l'/z$ for FSR. The radiator function in the collinear approximation takes the

simple, universal form $R_{coll} = \frac{\alpha}{2\pi} \log \frac{Q^2}{m_e^2} \left(\frac{1+z^2}{1-z} \right)$ so that the cross section is obtained

from $d\sigma_{ISR} = \int \frac{dz}{z} R_{coll} d\sigma_{Born}(l^\mu \rightarrow zl^\mu)$ (and similarly for FSR). The potentially large

logarithm $\log Q^2/m_e^2$ may reach the order of 10% at large Q^2 .

In the following we will discuss as example, results from the MC generator DJANGO [1]. The event generator DJANGO simulates deep inelastic lepton-proton scattering for both NC and CC events including both QED and QCD radiative effects. DJANGO contains the Monte Carlo program HERACLES and an interface of HERACLES to LEPTO. The use of HERACLES allows to take into account the complete one-loop electroweak radiative corrections and radiative scattering. The LUND string fragmentation as implemented in the event simulation program JETSET

is used to obtain the complete hadronic final state. At low hadronic mass, *SOPHIA* is used instead of *LEPTO*. *DJANGO* comprises the programs (formerly kept separately) *DJANGO6* and *HERACLES*. The interface is to version 6.5.1 of *LEPTO*. For eRHIC *DJANGO* was upgraded to use nuclear PDFs as available in *LHAPDF*. From version 4.6.10 on *DJANGO* simulates also longitudinal polarized deep inelastic lepton-proton scattering for both NC and CC events including both QED and QCD radiative effects. Figure 0-3 shows a diagram of the workflow in the MC code.

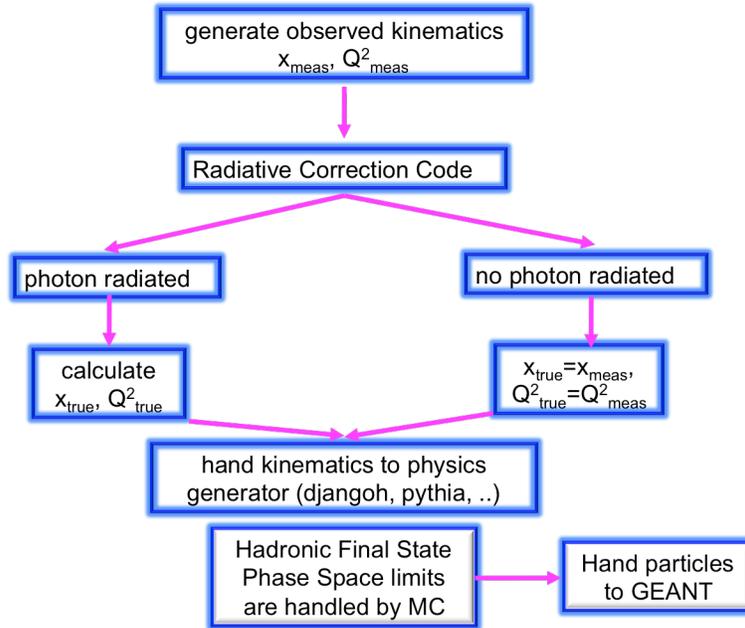


Figure 0-3: Workflow in the MC code *DJANGO*.

As an example, we show in Figure 0-4 the numerical results for the correction factor $r_c(y) = \frac{d\sigma / dy|_{O(\alpha)}}{d\sigma / dy|_{Born}} - 1$ for the structure functions F_2^{Au} (left) and $F_{2,cc}^{Au}$ (right) from $e+Au$ scattering with beam energies of 20 GeV on 100 GeV. The general features following from the preceding discussion are clearly visible: corrections are large at large y and small Q^2 , while corrections become negative at large Q^2 and small y . Requiring a hadronic final state as a charmed meson removes the elastic tail and the contribution from low-lying resonances. A similar effect can be achieved cutting on $E-p_z$ from the Jacquet-Blondel method. The reduction of the radiative corrections is considerable at largest y and at small Q^2 , but probably not yet sufficient at larger values of Q^2 and small y .

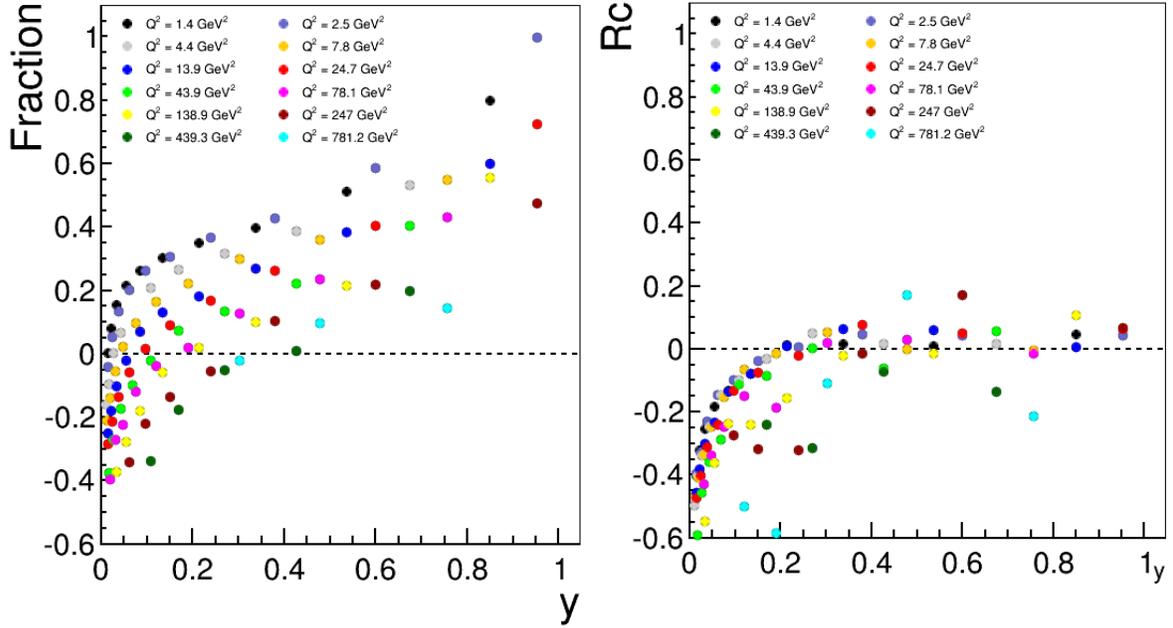


Figure 0-4: Radiative correction factor $r_c(y)$ for the structure functions F_2^{Au} (left) and $F_{2,cc}^{Au}$ (right) from e+Au scattering with beam energies of 20 GeV on 100 GeV.

Since the determination of the true structure functions requires an iterative unfolding procedure, it is important to study how the radiative corrections do depend on the assumed input structure functions, detector acceptance, responses and detection efficiencies. Especially as this effects are convolutions of each other and therefore don't factorize at all. Corrections due to the emission of photons from the hadrons, or quarks in the deep inelastic regime, require a careful separation into contributions which should be considered as a part of the hadron structure (leading to an electromagnetic contribution to scaling violations [2]) and contributions which can, in principle, be related to the observation of direct photons radiated from quarks. The interference of radiation from the lepton and the quark is small [3]. In certain phase space regions one may expect higher than one-photon corrections to be important. For example, soft-photon exponentiation will be necessary at small y and large x_B . The procedure is well known and straightforward. Finally, multi photon radiation may become important at large y and small x_B . In this case, the collinear approximation is sufficient to reach a precision at the level of one percent [4]. Many more studies on the described way to do radiative corrections can be found in [5].

Deliverables:

In the following the deliverables, which should be achieved at the end of the project should be achieved:

1. Calculate radiative corrections for transverse polarized observables to measure TMDs and polarized exclusive observables.

2. Provide proof that the MC phase space constrains on the hadronic final state is equal to calculating radiative corrections for each polarized and unpolarized semi-inclusive hadronic final state independently. A philosophy, which is currently still followed by some fixed target experiments and neglects the coupling between radiative corrections and detector effects [5].
3. Define a software framework and develop a library based on this framework, which integrates the radiative corrections depending on polarization and other determining factors in a wrapper-software, which allows the same integration routines for different Monte Carlo generators following the example of LHAPDF, which does this for different PDFs.

[1] For details see <https://wiki.bnl.gov/eic/index.php/DJANGO> and references given on this web-side

[2] H. Spiesberger, Phys. Rev. D52, 4936 (1995), hep-ph/9412286.

[3] H. Spiesberger et al., Contribution to Workshop on Physics at HERA, Hamburg, Germany, Oct 29-30, 1991.

[4] J. Kripfganz, H. Mohring, and H. Spiesberger, Z. Phys. C49, 501 (1991).

[5] Talk by E.C. Aschenauer at <https://www.jlab.org/conferences/radiative2016/program.html>

Validation and tuning of critical Geant4 physics in the energy regime of the EIC (Andrea Dotti)

Geant4 is the most widely used detector simulation toolkit for HEP&NP experiments. One of the characteristics enabling its success is the modular nature of the toolkit: different variants and options for the physics algorithms and a powerful geometry package allow to simulate different setups, from the geometry and physics of the typical collider experiment to the international space station to the human DNA.

We foresee that Geant4 will be extensively used for EIC simulations because it is well known by the NP community and because a large set of applications based on it already exists.

While the EIC detector design can rely on the existing experience to implement the EIC specific setups (for example via the GDML interface), there are some aspects that we think fit properly in the context of the EIC SW R&D consortium: Geant4 comes with a large collection of algorithms that need to be assembled in *physics lists* to cover the range of energy and particle types of interest. These physics lists need to be validated for the specific setups of EIC.

We plan to capitalize on the physics lists that have shown the best results at existing NP and HEP experiments and use them as a starting point for further development. We plan to perform a set of tests of the available options and variants to effectively identify the most important differences between them. In some cases we will also need to evaluate the CPU cost of some physics options considering the general CPU budget of EIC simulations. A non-exhaustive list of questions we plan to address in this study is:

- Do we need to enable, to improve the simulation of hadronic interactions, the high-precision neutron treatment, what is the CPU increase in such a case?
- Geant4 offers three different intra-nuclear cascade model for the energy regime 100MeV ~ 10GeV, each one has some strength and weakness, which is the most adequate for the typical EIC energy and particle species?
- Do we need to tune some transition regions between models to obtain a smoother dependence of observables as a function of primary energy?
- Are the detector technologies foreseen at EIC included in the Geant4 validation test suite (e.g. materials, physics processes)?
- What is the role of the simulation of secondary ions interactions in EIC detectors? The Geant4 collaboration, using mainly HEP derived data, that are less sensitive to this aspect, may need help to extend its test-suite to include such interactions

For this study, in collaboration with the Geant4/SLAC team, we propose to use three exiting applications (already developed by the SLAC/Geant4 team):

1. The *SimplifiedCalorimeter* application will be used to verify shower shapes in calorimeters
2. The *ProcessTest* will be used to study interactions in thin detectors (e.g. inner detectors)
3. The *HepExpMT* application will be used, via a GDML interface, to study CPU performances on semi-realistic setups.

We plan to contribute in improving the applications and introducing the specific modifications needed for the validation of the physics lists of interest to EIC.

The final goal of this study is to define one or more physics lists recommended for the simulation of EIC detectors.

Deliverables

During the first year of the R&D we plan to:

1. Review the current validation strategy of Geant4 identifying what are the EIC specific interests that are currently not covered. We will identify which of the data-sets could be used to extend the Geant4 validation test-suite particularly fit to EIC energy/interactions. We plan to feedback these findings to the Geant4 Collaboration and eventually collaborate with experts to address these issues.
2. Extend the validation applications to address the EIC specific needs:
 - Develop simulation and analysis macros for *SimplifiedCalorimeter* and *ProcessTest* to generate and study the interactions of most interest for EIC
 - Evaluate a GDML-based simplified setup, to be used with the *HepExpMT* application, to measure CPU time-consumption of alternative physics list

Start the development of an universal event display for MC events (Sergei Chekanov)

One area of our activity will be focused on creating a generic event display for viewing generated (and detector reconstructed) events on web browsers. Besides clear outreach component of this project, it will allow validation of the EIC simulations, as well as comparison of different detector designs using an unified approach. Such web-based event displays already exist for other experiments. We have started to look into the following projects:

- There exists a web-based event display for the CMS detector [3]. It is based on the common WebGL technology and requires the OBJ file format to view the CMS detector geometry and pp-collision events. A development of a convertor from the popular geometry file formats used for particle experiments, such as GDML and HEPREP, to the OBJ files for a web-browser based on WebGL, could be an important direction towards comparisons of different detector designs and technologies, and for viewing the generated physics processes for validation checks.
- FNAL's Scientific Computing Division (SCD) is focusing on using an open-source application called ParaView [2] for their experiments on the intensity frontier. ParaView is widely used at DOE Supercomputing centers and embodies the state-of-the-art in visualization science. FNAL SCD found that it has the fastest rendering and interaction times and some very unique features such as overlaying detector geometries with CAD drawings. ParaView does have a web component, ParaViewWeb, that uses WebGL [3], but FNAL SCD has not yet tested it. They are discussing with the Geant4 group at SLAC on how to provide a Geant4 visualization plugin for ParaView.

In FY17, we will evaluate how the CMS and ParaViewWeb event displays can be used for the existing software frameworks and the HepSim MC repository we are working on (page 17).

[1] <http://ispy-webgl.web.cern.ch/ispy-webgl>

[2] <http://www.paraview.org>

[3] <http://www.paraview.org/web/>

Promote open-data developments for efficient data-MC comparisons (Stefan Prestel)

For the software R&D for the EIC, it is beneficial to adapt successful solutions from other fields of physics, in particular from high-energy physics. Many things can be learned from the infrastructure available at the LHC for data-MC comparisons and MC tuning:

Experiment and theory collaborate by sharing information through well-defined channels. Experiment presents data in the form of tables/plots. Given knowledge of the experimental analysis objects, these can be used to compare theory against data by using analysis frameworks. In order to do this, it is necessary to have theory tools that are able to provide plots using as much detailed information as used to perform the experimental analysis. If such *theory* tools are public, then an experimental analysis can be scrutinized prior to publication, and analysis strategies can be improved to suppress backgrounds.

This leads to a mutually beneficial cycle, in which experimental measurements challenge the predictions of theory. The theory tools then improve, which then allows even more refined analysis strategies, leading to even more interesting/challenging data. If this works as intended, then challenging data will lead to new insights, which then gets incorporated into the theory tools. In this sense, up-to-date and maintained theory tools serve as a repository of our knowledge.

For interesting data to be transferred to our understanding of physics efficiently, it is crucial that the tools that allow this transfer are public, easy to use, well-maintained, and that known results are easily accessible. If this is the case, then a large community can help in facilitating this knowledge transfer. If it is difficult or daunting to compare theory to data, then insights will develop much slower.

To not unnecessarily delay progress, it is thus paramount to encourage and promote open-source/open-data developments. This includes public databases for experimental data, like the already well-established HepData web page [1, 2, 3]. Open-source analysis frameworks that can be used by - and allow contributions from - both experimental and theoretical physicists can help accelerate progress. An example of this is the Rivet software [4], which enables quick analysis prototyping for experiments as well as straightforward data-to-theory comparisons. The advent and superb performance of the Large Hadron Collider has led to a strong push towards open-source developments in the High-energy physics community. This e.g. includes open-source software to adjust (i.e. *tune*) theory unknowns to a large variety of data in order to obtain theory models that can with more confidence be used to assess new measurements [5,6].

Open-source and open-data frameworks rely on detailed and universal interfaces, which have been accepted by the community (see e.g. [7,8]). The success of such software is intimately linked to usability and accessibility. The latter can be achieved by making results directly available in an easy-to-use web database, such as e.g. mcplots.cern.ch [9] or <http://atlaswww.hep.anl.gov/hepsim> [10]. Such truly public databases also provide excellent outreach opportunities, see e.g. <http://lhcatome.web.cern.ch> [11].

These lessons should be kept in mind when planning a successful EIC software program. On the one hand, this means that experimental data should be accessible, and analyses should be reproducible without effort. A flexible open-source software framework is essential for this task. This furthermore ensures not only data preservation, but also analysis preservation - the latter possibly being even more important, as a failure to do this leads to loss of important knowledge. On the other hand, it is also crucial that theory tools mature to become flexible *theory* knowledge repositories. This would entail software that is able to answer a very wide spectrum of physics questions, while still remaining extendible. Monte Carlo Event generators [12, 13, 14, 15, 16] provide excellent opportunities to act as theory tools for EIC physics.

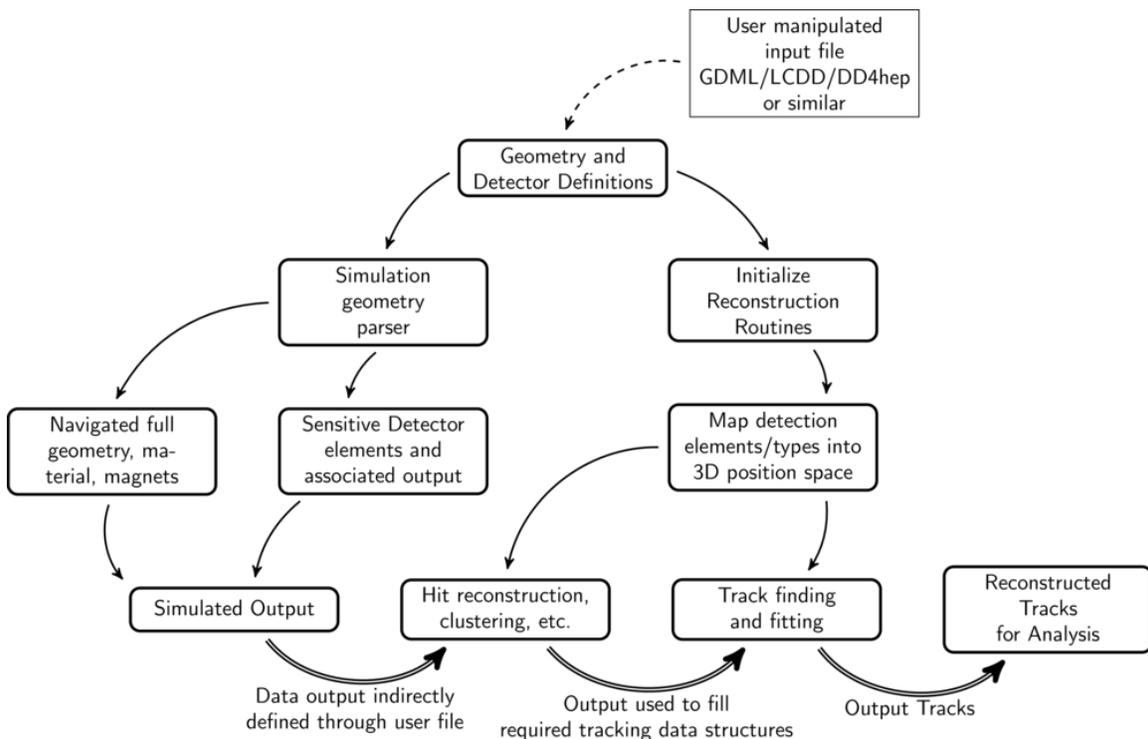
As part of our R&D effort, we will evaluate how tools for data-MC comparisons and MC tuning can be best provided for the EIC. This effort will result in a white paper. We will provide a tutorial how to use the RIVET [4] for analysis development, and how to use the PROFESSOR extension [6] as a tool to perform necessary Monte Carlo tuning. Given the on-going work on a library for radiative effects (see page 5) and the HepSim repository (see page 19), we will start this by using Djangoh [17] as example. We expect that a clear tutorial will help with analysis development, resulting in an accelerated pace of MC development, both concerning theory improvements, as well as tuning to existing data.

- [1] <http://hepdata.cedar.ac.uk/>
- [2] <https://inspirehep.net/record/289771>
- [3] <https://inspirehep.net/record/856996>
- [4] <https://inspirehep.net/record/847552>
- [5] <https://inspirehep.net/record/424112>
- [6] <https://inspirehep.net/record/825971>
- [7] <https://inspirehep.net/record/553387>
- [8] <https://inspirehep.net/record/725284>
- [9] <https://inspirehep.net/record/1238614>
- [10] <https://inspirehep.net/record/1285237>
- [11] <https://inspirehep.net/record/1125350>
- [12] <https://inspirehep.net/record/884202>
- [13] <https://inspirehep.net/record/712925>
- [14] <https://inspirehep.net/record/373072>
- [15] <https://inspirehep.net/record/685829>
- [16] <https://inspirehep.net/record/538940>
- [17] <https://wiki.bnl.gov/eic/index.php/DJANGO>

Work towards a common geometry and detector interface (Whitney Armstrong)

Defining the geometry and detection elements is critical for any simulation. Disseminating the exact parameters defining these elements to all aspects of simulation, reconstruction, and analysis is a non-trivial problem. A simplified process where all the geometry and detector information is localized into one source, i.e., a text file written in a markup language, is shown in the figure below. These definitions then need to be shared between the simulation and reconstruction tasks. In addition to navigating particles through the full geometry, materials, and fields, the simulation also must have a detailed description of the sensitive detection elements and their resulting data structures. The output from the simulation, like real data, does not directly encode the high-level analysis objects such as clusters and tracks. Therefore, the same geometry and detector definitions have to be shared among all the reconstruction routines in order to properly determine the track positions in 3D space.

A common geometry and detector interface is not limited to just parsing a text file. It involves building an interface to the logical objects represented in the file and providing generic methods to extract the position information crucial for a general purpose unified track reconstruction suite as discussed below. We propose to develop these tools together and, more importantly, provide detailed



documentation on how they fit together in the landscape of simulation and reconstruction software.

It should be noted, that there exists a similar working concept of sharing the essential details of the sensitive volume information between simulation and reconstruction codes developed from scratch within the EicRoot software framework. The respective library is ROOT-based and keeps all the necessary information in the same binary TGeo files, which contain the detector actual geometry description.

A similar concept is also used in the SLIC software framework. We plan on benefiting from the lessons learned from and build upon the success of these projects.

Deliverables:

1. Define and document a flexible common geometry and detector definition interface.
2. Develop tools to use the interface with an emphasis on feedback from users.
3. We will develop a preliminary library in order to complete a feasibility study in conjunction with the unified track reconstruction discussed below.

The listed deliverables completed together will allow the detector simulation and design feedback loop to be completed with ease, leading to excellent and sophisticated detector designs.

Work towards an unified track reconstruction (Alexander Kiselev)

Partial unification of the simulation and data analysis tools at this early stage of the project when 1) the manpower is limited and 2) there already exists a *de facto* diversity of the EIC-related software frameworks developed to a certain extent at different sites is extremely important. One of the option which we consider is to develop a suit of the common track reconstruction tools.

The main objective here is that the EIC detector layout in general (and tracking subsystem in particular) is well defined in a sense, that it comprises a low mass and moderate ($\sigma P/P$ of an order of 1% or better) momentum resolution tracker with a full geometric coverage in the pseudo-rapidity range of at least $[-3.5 .. +3.5]$. A typical EIC detector would consist of a relatively small *barrel* silicon vertex tracker with a high spatial resolution, a number of forward and backward *discs* potentially based on a similar technology, a *volume* tracker (like a TPC or a drift chamber) at central rapidities and perhaps a set of complementary detectors with fast timing response (like GEMs), all in a strong solenoid magnetic field.

A typical physics analysis data flow (from GEANT-based simulation of a particular set of physics events to the hit digitization, event reconstruction and further on to the user analysis) should also be the same or very similar in all site-specific software frameworks.

It must therefore be possible, despite the variety of the presently used EIC simulation frameworks, either purely GEANT4-based (GEMC@JLAB, fun4all@BNL, SLIC@ANL) or the virtual Monte-Carlo ones (EicRoot@BNL) to decide on a unified data format at some intermediate stage (presumably, after the digitization) and make use of a common library of *EIC tracking tools*. The tool set itself can consist of (but not necessarily be limited to) a track finder code, perhaps logically split between forward and central rapidity regions, optimal track fitting algorithms, extrapolation to outer detector locations (like RICH and/or calorimeters), vertex finder and fitter, beam line constraint accounting code, kinematic fitting, etc. Part of the effort must be spent on providing interfaces to existing modular packages, like Millipede [1] for the detector misalignment studies or RAVE [2] for vertexing. The other codes must either be written from scratch for the anticipated EIC detector geometry or be ported when appropriate from other similar applications (like FairRoot clones).

Output of the unified track reconstruction library call can be piped back into the specific software framework data flow, which would naturally require a standardization of the respective I/O format as well.

No matter which approach is taken for a particular part of the outlined project, it will require months of physicists and/or software experts work with a clear benefit of having at the end one shared solid code set instead of a few lousy ones. Such a

scheme should also simplify the eventual migration to the “final EIC software framework(s)” once the site selection is made and physics collaboration(s) start to take over.

It should be noted specifically, that the success of the project will to a large extent depend on our ability to provide a unified access calls to the geometry database of a particular framework (material distribution, magnetic field, detector 3D locations). Therefore it is important that this work is closely tied to the common geometry and detector definitions proposed above.

Deliverables:

1. Based on the outcome of the common geometry definition exercise and the Monte-Carlo generator output format selection we will perform a detailed feasibility study of extracting existing track reconstruction codes (either implemented in EIC-related frameworks already or available as standalone packages) into a consistent set of tools (i.e. a library), which can then be used as a core tracking engine within any of the existing EIC software environments, complying with the agreed upon data structures and interfaces to a common geometry.
2. Provided the outcome of the feasibility study is positive we will consider starting the actual portable EIC tracking library implementation from the highly configurable genfit-based track fitting code, comparable in the provided functionality to the already available EicRoot tracking R&D toolkit.

[1] arXiv:hep-ex/0208021

[2] Journal of Physics: Conference Series 119 (2008) 032037

Developing interfaces to forward compatible, self-descriptive file formats (Sergei Chekanov)

Monte Carlo simulations for the EIC require integration with modern data formats. This is necessary for long-term maintainability of simulated data, and for effective comparison of the EIC results with the existing tools used by a large particle-physics community. Significant number of Monte Carlo generators for ep collision events has been developed in the past for the HERA experiments. They typically have outdated interface for output files (developed 15-20 years ago). To mitigate this problem, we propose to modernize the persistency framework for the EIC Monte Carlo generators.

The proposed step in this direction is to interface EIC generators with the ProMC file format [1]. This will allow generated EIC simulated samples to be stored in the HepSim database [2] in order to take advantage of the Open-Science Grid (OSG) for data storage, long-term preservation and processing. This will also enable easy conversions to the ROOT format, processing data using fast detector simulations, or other complex analysis frameworks. The ProMC library can easily be deployed on high-performance computers (HPC), and due to its compactness (ProMC files are 30% smaller than corresponding ROOT ones), can provide an effective persistency framework with small footprint on data input and output.

We also would like to develop a more suitable event layout for storing EIC data. For example, a typical ep/eA collision events require a certain number of parameters which characterize ep/eA events, such as Q^2 , Bjorken x , W , etc.. Thus, the generic layout of data used by the ProMC files should be adjusted for storing ep/eA - specific event characteristics. The already existing *ep-smear* package [3] developed by BNL EIC taskforce can be a good starting point in this direction.

Deliverables:

In FY17, we will setup a HepSim repository for the EIC and provide guidelines for using it for the EIC simulations. We will evaluate the requirements for a EIC data format and will use the common ROOT format and the modern ProMC format as baseline for comparison to future data formats. We will start designing the EIC data format by maintaining a list of variables for the various ep/eA processes.

[1] S.Chekanov, E.May, K. Strand, P. Van Gemmeren, ProMC: Input-output data format for HEP applications using varint encoding, ANL-HEP-PR-13-41, [arXiv:1311.1229](https://arxiv.org/abs/1311.1229), Computer Physics Communications 185 (2014), pp. 2629-2635

[2] HepSim web-based repository for Monte Carlo simulations.
<http://atlaswww.hep.anl.gov/hepsim/>

[3] <http://www.star.bnl.gov/~tpb/eic-smear/annotated.html>