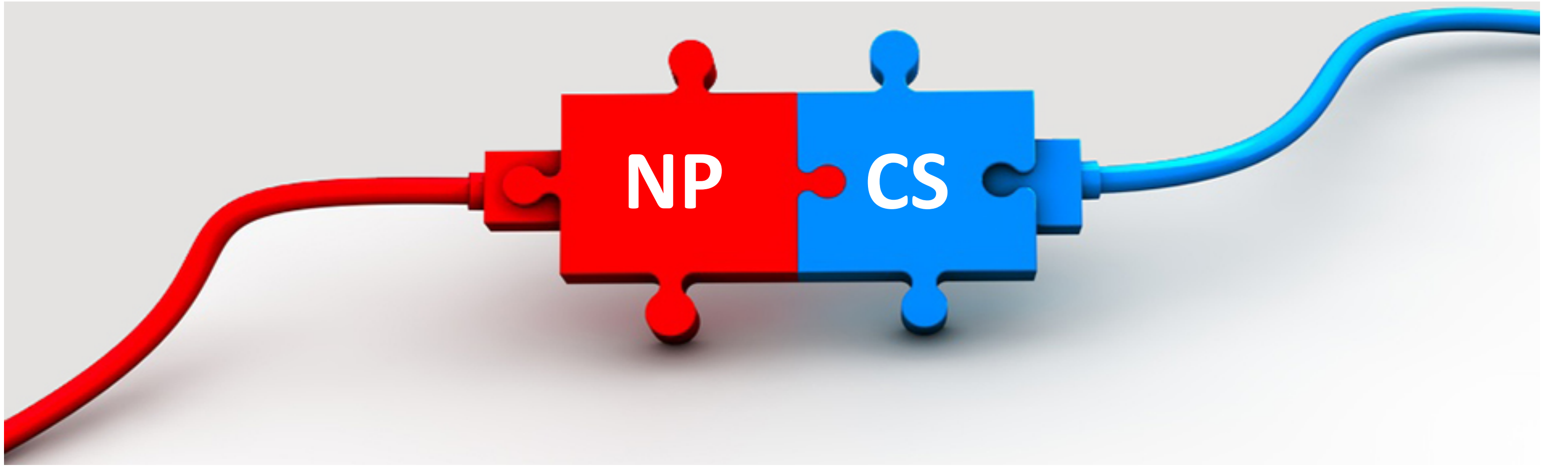
Computing for the Electron-Ion Collider

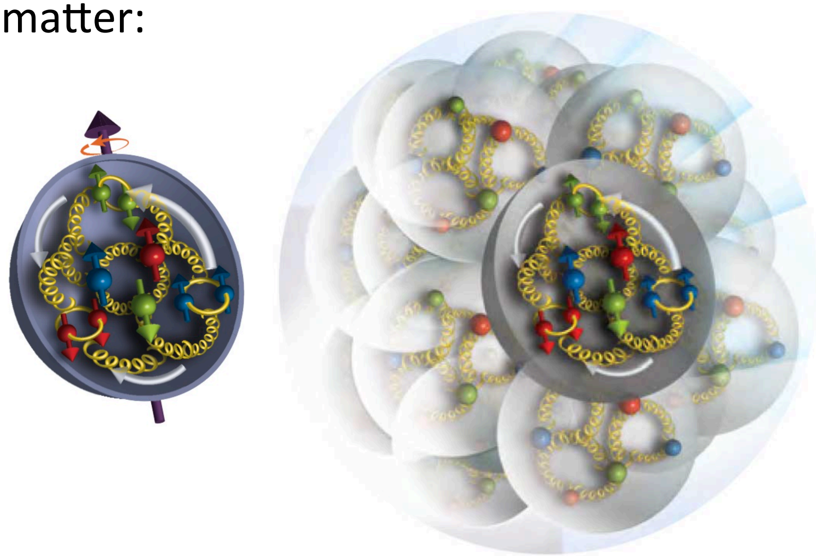
Markus Diefenthaler (Jefferson Lab)



22nd International Conference on Computing in High Energy and Nuclear Physics

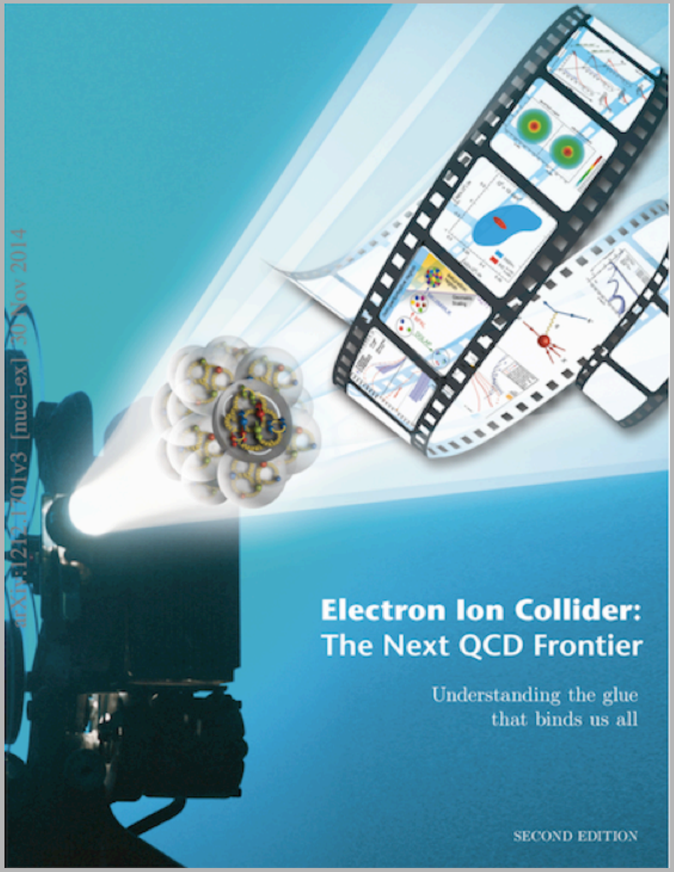
Electron-Ion Collider (EIC)

- next-generation U.S. facility to study quarks and gluons in strongly interacting matter:



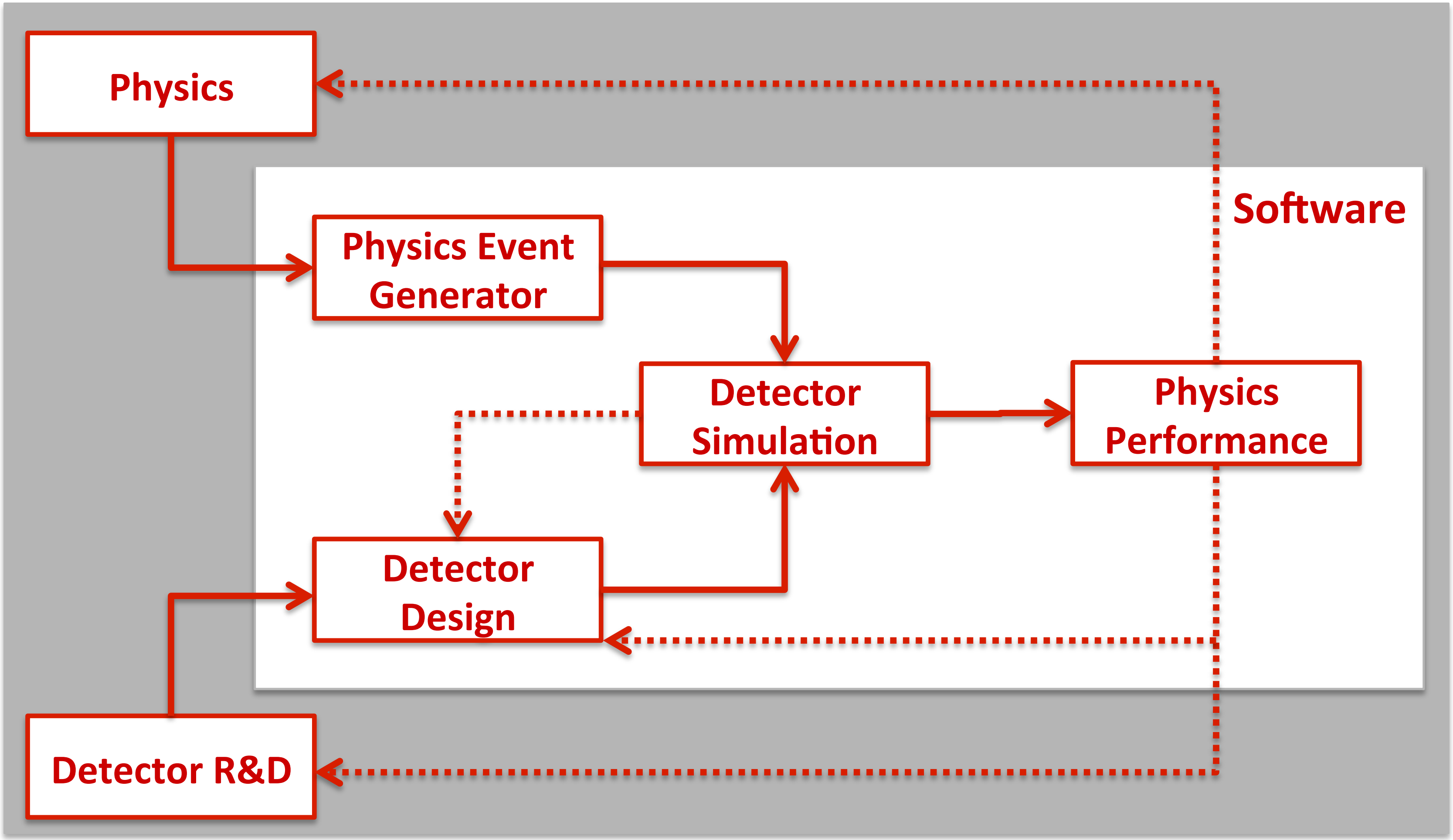
- world's first collider of:
  - polarized electrons and polarized protons/light ions
  - electrons and nuclei

EIC: The Next QCD Frontier



Eur.Phys.J. A52 (2016) no.9, 268

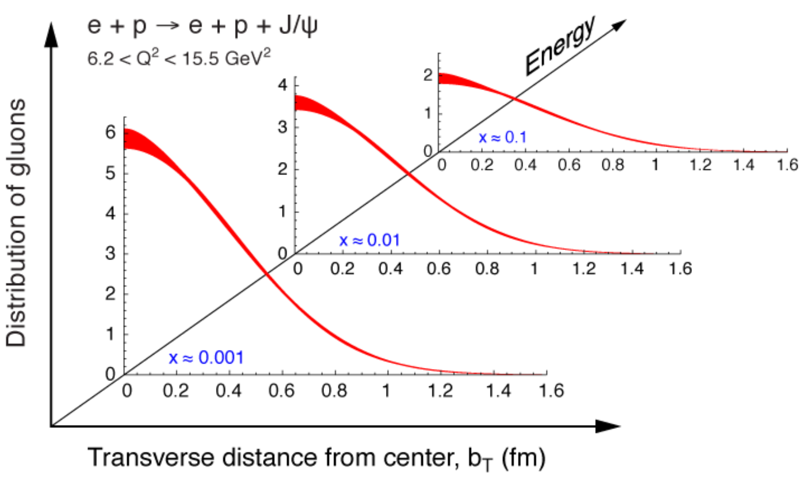
EIC R&D and software development



Computing Challenges in NP

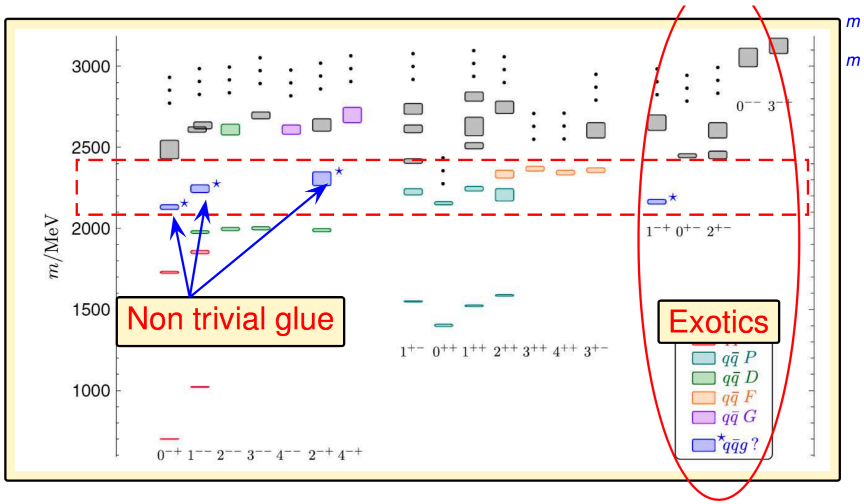
NP experiments driven by beam intensity, polarization, exquisite control of background and systematic

multi-dimensional, e.g., 3D imaging of quarks and gluons



high statistics in five or more dimensions and multiple particles

multiple channels, e.g., discovery search of gluon-based exotic particles (PWA, 1000s of waves)



strongly iterative analysis for reliable, model-independent analysis

Analysis environments

Developments of analysis environments:

- new projects starting (JLab 12 GeV) and on the horizon (EIC)
- likely explosion of data even at the small nuclear experiments
- think about the **next generation(s) of analysis environments** that will **maximize** the **science output**

LHC experiments: tremendous success in achieving their analysis goals and producing results in timely manners

Lesson learned at LHC experiments:

- as the complexity and size of the experiments grew
- the complexity of analysis environment grew
- time dealing with the analysis infrastructure grew

Anecdote from LHC
a typical LHC student or post-doc spends up to 50 % of his/her time dealing with computing issues

New analysis environments

User centered design

- understand the user requirements first and foremost
- engage wider community of physicists in design whose primary interest is not computing
- make design decisions solely based on user requirements
- web-based user interfaces, e.g. interactive analysis in Jupyter Notebook

Future compatibility (both hardware and software)

- most powerful future computers will likely be very different from the kind of computers currently used in NP (Exascale Computing)
- structures robust against likely changes in computing environment
- apply modular design: changes in underlying code can be handled without an entire overhaul of the structure

Think out of the box

- the way analysis is done has been largely shaped by kinds of computing that has been available
- computing begins to grow in very different ways in the future, driven by very different forces than in the past (Exascale Computing)
- think about new possibilities and paradigms that can and should arise

Big Data - A possible paradigm shift for NP / HEP?

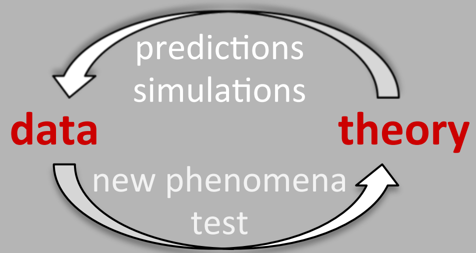
- **Big Data** is not about size
- **Big Data** is about the ability to quickly analyze large amounts of data. i.e.
  - have all raw and processed data permanently store
  - in a scale-able random access storage
  - with fast, efficient data indexing (lookup) capabilitie
  - ➔more efficient use of computational resources
  - ➔fast data (re)processing and analysis

- statistical language of **R**:
  - driven statistical analysis for more than a decade
  - emerging as the leader in statistical languages for **Big Data**
  - an alternative to ROOT?
  - NP should acquire some knowledge here (cooperate with other fields / industry)

- **NoSQL** (non-relational) databases:
  - more flexible
  - better scaleable than traditional, relational databases
  - e.g., a graph database (e.g., used by Facebook)

Interplay of data and theory

Feedback loop between data and theory



Comparison to:

- analytical calculations
- **Monte Carlo (MC) simulations**
- **Lattice-QCD calculations**

Data-theory comparison: relies on

- open access to data-theory tools
- standardization of data-theory tools
- comparison tools for quick turnaround

MC event generator:

- faithful representation of QCD dynamics
- based on QCD factorization and evolution equations

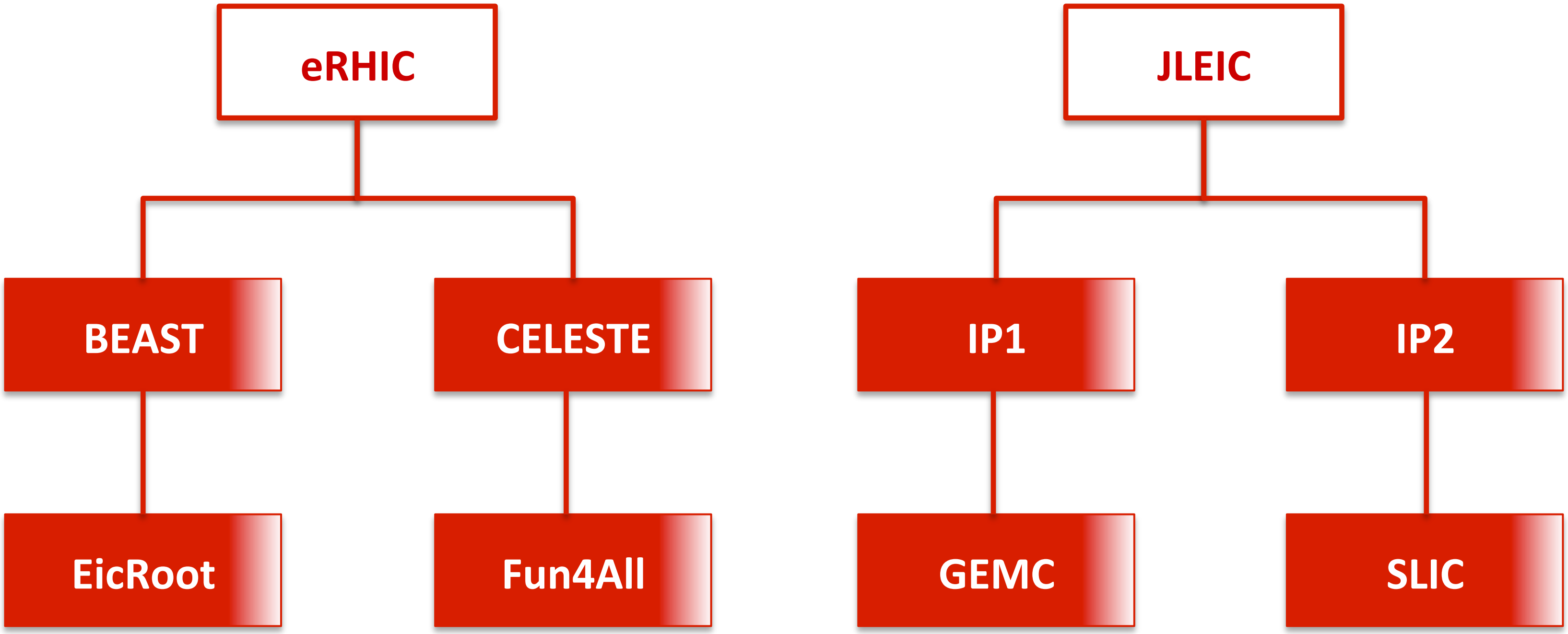
Usage by experimentalists:

- detector corrections
- analysis prototyping
- comparing to theory

Usage by theoreticians:

- easy off-the-shelf state-of-the-art tool that looks like data
- validate against and investigate theoretical improvements

Existing software frameworks for the EIC



Building on existing EIC software:

- build forward-compatible interfaces between existing frameworks / tools
- identify common tools and improve them (e.g. MCEG)
- add tools that are forward-compatible with existing frameworks

Forming a software consortium for the EIC

September 2015 EIC Software Meeting
Workshop organized by Elke-Caroline Aschenauer and Markus Diefenthaler
https://www.jlab.org/conferences/eicsw/
review of existing EIC software frameworks and MCEG available for the EIC

January 2016 Generic R&D Meeting: **LOI for Software Consortium**

> **Review** "A robust software environment, compatible with the existing software frameworks, is very important for the development of the physics case for the EIC."

March 2016 Future Trends in NP Computing
Workshop organized by Amber Boehnlein, Graham Heyes, and Markus Diefenthaler
https://www.jlab.org/conferences/trends2016/
discussion of computing trends, e.g., Big Data, machine learning, Exascale Computing
incubator for ideas on how to improve analysis workflows in NP

July 2017 Generic R&D Meeting: **Proposal for Software Consortium**
consisting of scientists from ANL, BNL, JLab, INFN Trieste, and SLAC
R&D funds for workshop, travel, and students have been awarded (eRD20)

Global objectives

Interfaces and integration

- connect existing frameworks / toolkits
- identify the key pieces for a future EIC toolkit
- collaborate with other R&D consortia

Planning for the future with future compatibility

- workshop to discuss new scientific computing developments and trends
- incorporating new standards
- validating our tools on new computing infrastructure

Organizational efforts with an emphasis on communication

- build an active working group and foster collaboration
- documentation about available software
- maintaining a software repository
- workshop organization

Immediate development in FY17

Interfaces and integration

FY17

- start the development of a library for simulating radiate effects
- work towards a common geometry and detector interface
- work towards an unified track reconstruction
- collaborate with **TMD MC** and **DPMJetHybrid** (eRD17) and other software projects that are essential for an EIC

Planning for the future with future compatibility

FY17

- validation of critical Geant4 physics in the energy regime of the EIC
- start the development of an universal event display for MC events
- promote open-data developments for efficient data-MC comparison from the beginning
- build interfaces to forward compatible, self-descriptive file formats

Organizational efforts with an emphasis on communication

FY17

- build a community website
- organize software repositories dedicated to the EIC
- organize a workshop

Computing for the Electron-Ion Collider

- **Electron-Ion Collider** (EIC): next-generation U.S. facility to study quarks and gluons in strongly interacting matter
- **NP experiments**: driven by precision to access the **multi-dimensional** and **multi-channel** problem space
- **EIC Computing**: think about the **next generation(s) of analysis environments** that will **maximize** the **science output**
- **EIC Computing consortium**:
  - interfaces and integration
  - planning for the future with future compatibility
  - organizational efforts with an emphasis on communication