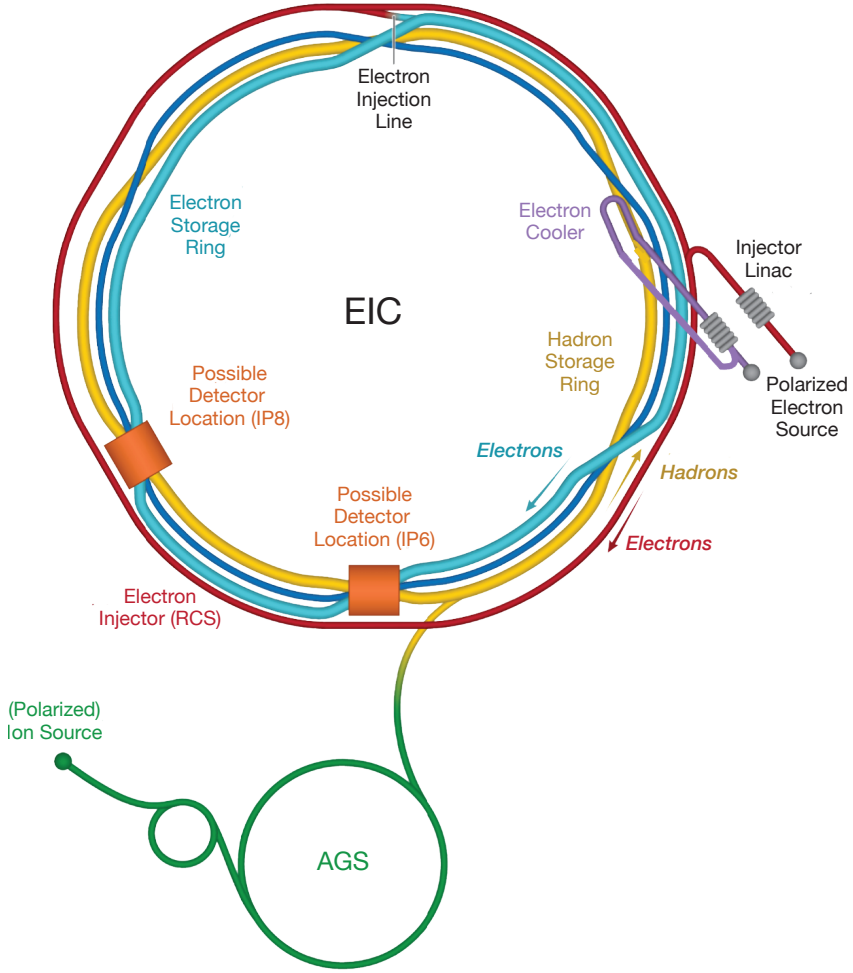
EIC SOFTWARE:
Statement of Principles

Markus Diefenthaler

EIC Center at Jefferson Lab

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 are relevant.
- BNL and Jefferson Lab will be host laboratories for the EIC Experimental Program. Leadership roles in the EIC project are shared.

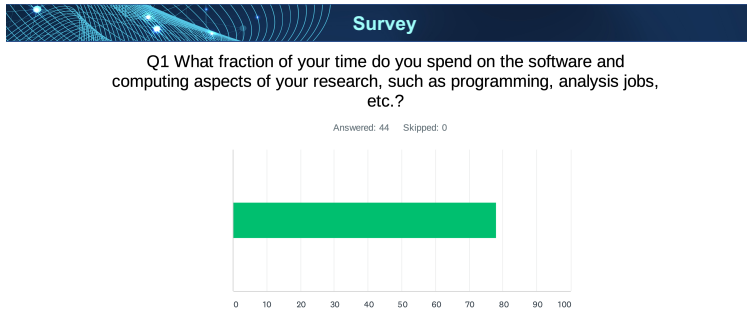
Lessons Learned About EIC Software

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: Computing & Software and Simulation, Production, & QA Working Groups
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 EIC scientists of all levels worldwide can participate in EIC analysis actively.
- **User-Centered Design**: To achieve this goal, we must develop simulation and analysis software using modern and advanced technologies and 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** with AI at the DAQ and analysis level.

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:

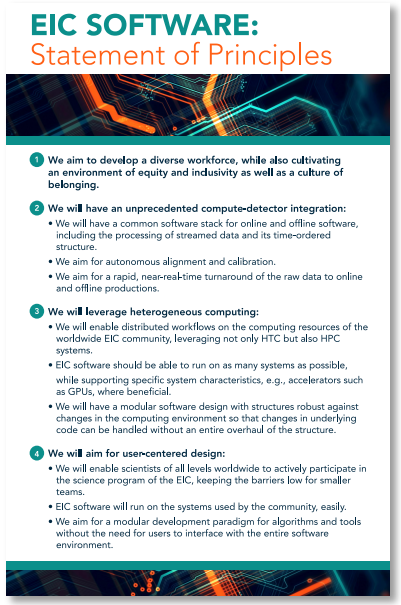
- **Yellow Report**: Physics case, the resulting detector requirements, and the evolving detector concepts for the EIC:
  - Mainly fast simulations and full simulations of detector components.
  - Foundation for detector collaboration proposals.

- **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

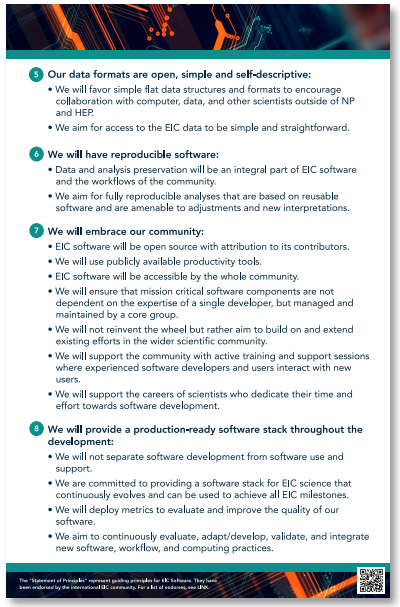
- **How to decide on our software stack?**
  - How do we ensure we work towards to our vision for EIC Software?
  - How do we ensure we meet the needs of the EIC community?

- **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 indusivity 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.

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.
- 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.

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.
- 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.

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.

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

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.
- 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:

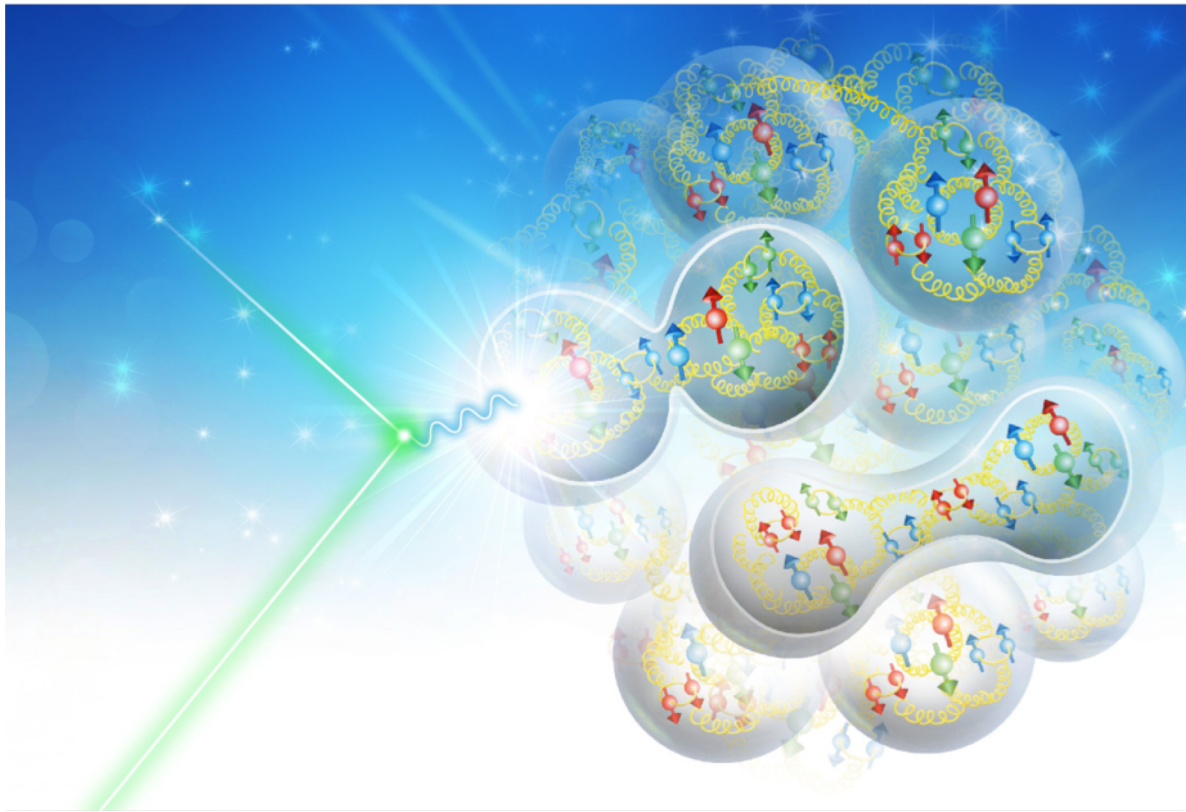
- We will not separate software development from software use and support.
- We are committed to providing a software stack for EIC science that continuously evolves and can be used to achieve all EIC milestones.
- We will deploy metrics to evaluate and improve the quality of our software.
- We aim to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

EIC Software: Statement of Principles

Markus Diefenthaler

mdiefent@jlab.org

- **Vision** for EIC Software & Computing:
  - User-Centered Design to ensure that EIC scientists of all levels worldwide can participate in EIC analysis actively.
  - Rapid turnaround of data for the physics analysis and to start the work on publications.
- The **Statement of Principles** represent **guiding principles** for EIC Software (Webpage, PDF version)
- The Statement of Principles have been **endorsed by the international EIC community**.



- **Special thanks to**: Amber Boehnlein, Andrea Bressan, Wouter Deconinck, Rolf Ent, Jin Huang, Sylvester Joosten, David Lawrence, Graeme Stewart, Torre Wenaus, Rik Yoshida and of course the EICUG SWG and EPIC Computing & Software and Simulation, Production, & QA Working Groups.