EIC Software Infrastructure Review

EPIC

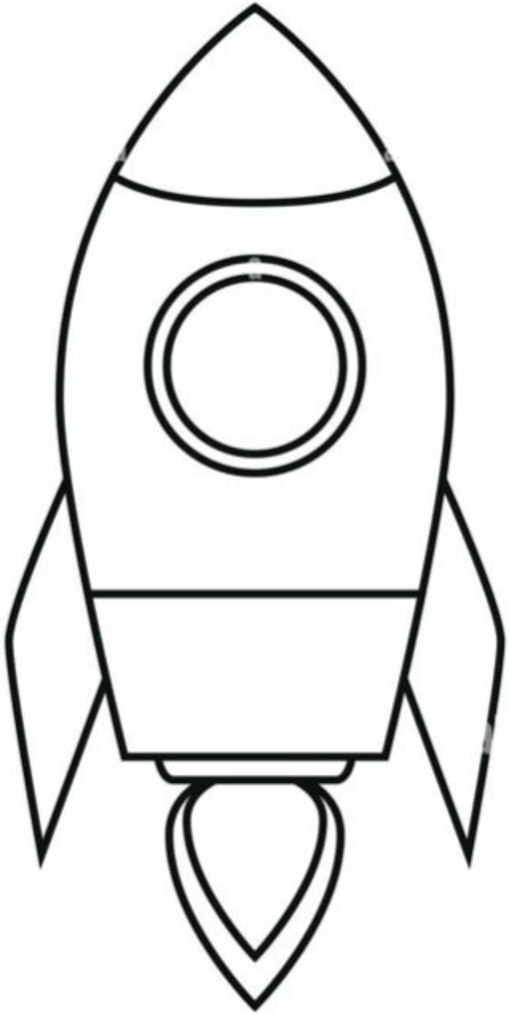
Introduction

Sylvester Joosten

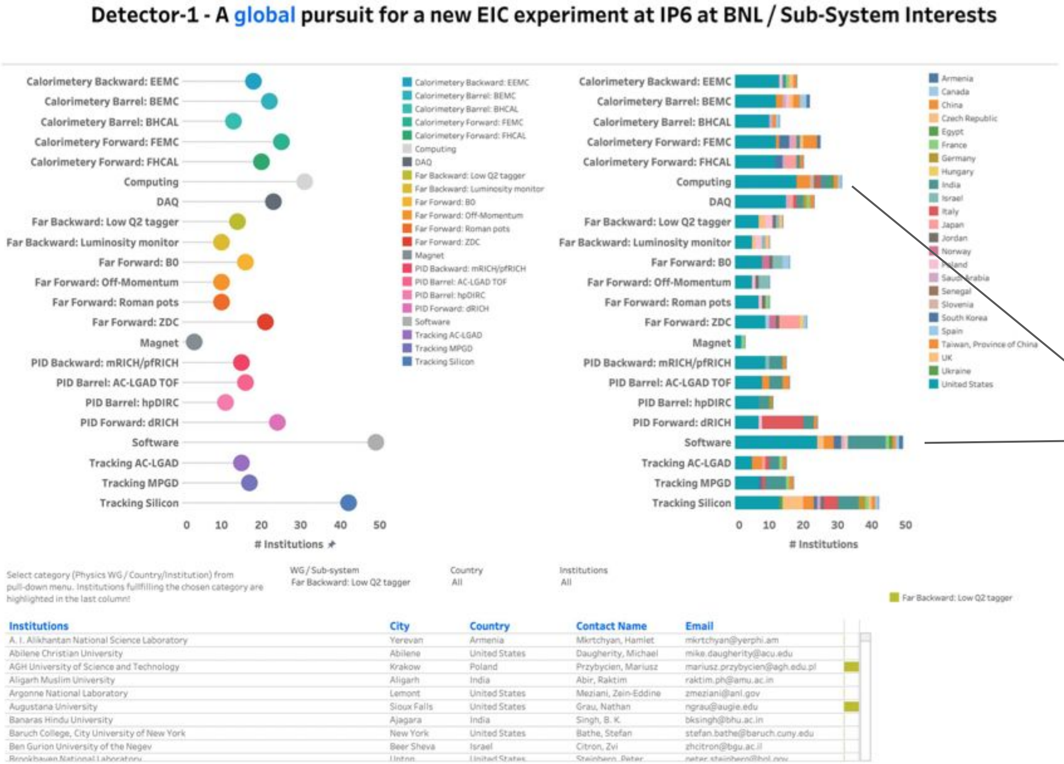
On behalf of the EPIC Collaboration

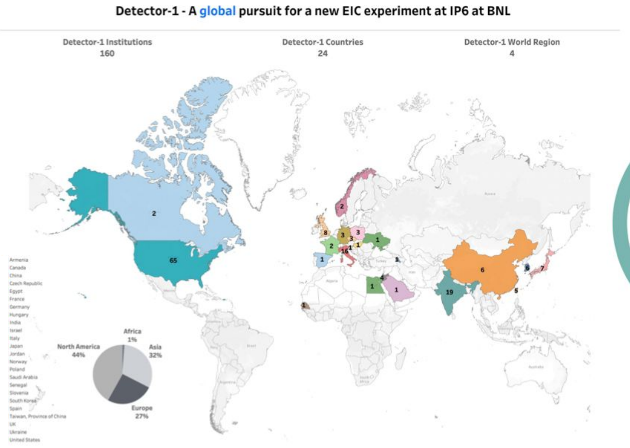
Outline

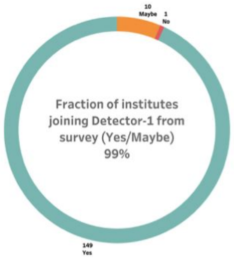
1. Lessons learned from years of EIC software efforts
2. EIC Software Statement of Principles
3. Towards a unified EPIC software strategy
4. Introduction of the toolkit components
5. What about detector 2?
6. Summary



We are a large collaboration (160 institutions)







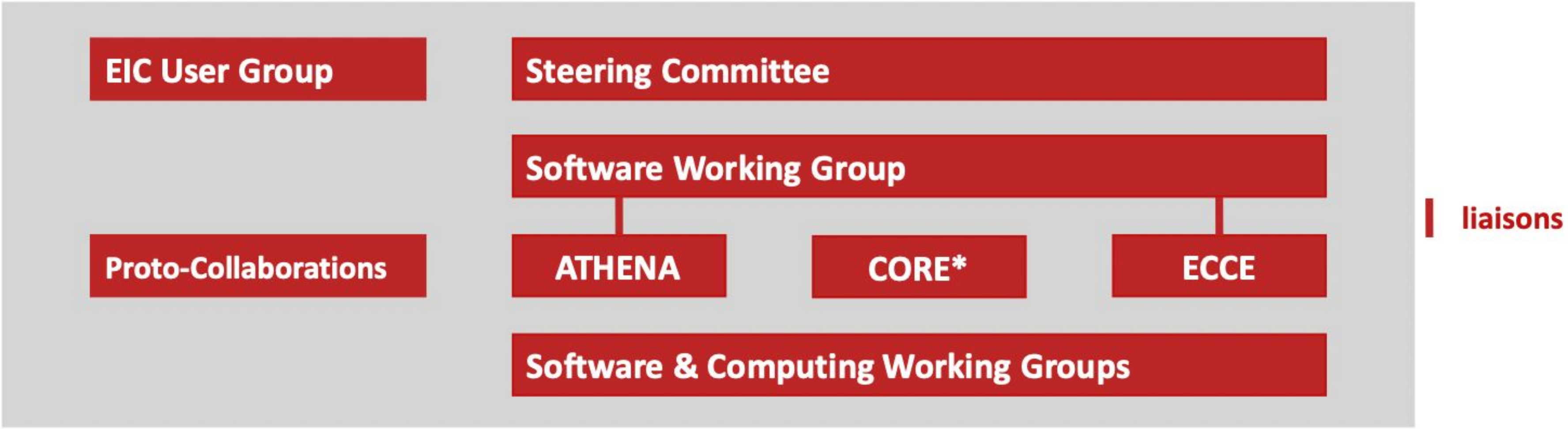
49 groups indicate commitment to work on software for EIC, of which 22 also want to contribute to computing

More commitments to software than *any* other topic!

Lessons learned from years of EIC simulations

- We are not starting from scratch, we have years experience from the EIC Software Consortium, the EICUG SWG, and doing EIC simulations for the Yellow Report and the Detector Proposal
- Both the ATHENA and ECCE proto-collaborations have successfully conducted detailed full detector simulations
    - ATHENA successfully built and deployed a prototype modular software stack based on modern common NHEP packages
    - ECCE leveraged familiar NP software to ensure all milestones could be reached, with the intent to reevaluate the software stack going forward
- The EICUG organized a series of "Lessons Learned" workshops to prepare the community to evolve from the proposal period towards software and computing for EIC itself.

Software is a community effort

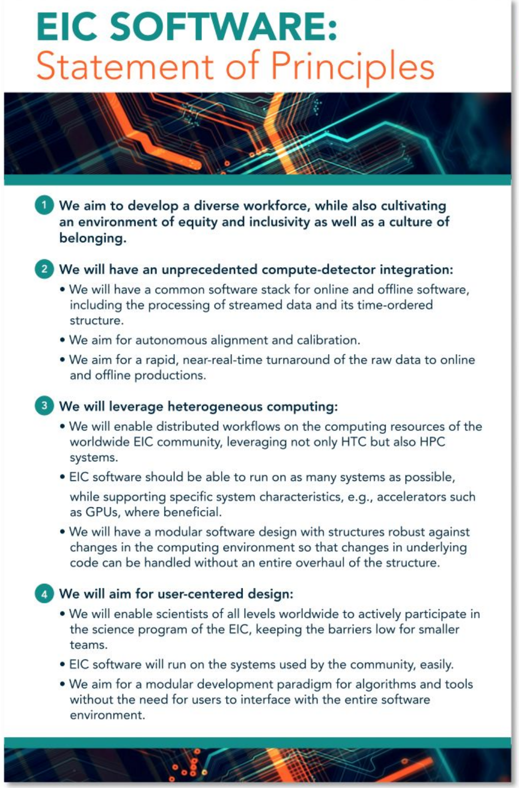


* CORE adapts existing software for their needs and has a far smaller software effort than other proto-collaborations.

The “Lessons Learned process effectively started the process of unifying the different software and computing efforts, laying the groundwork for the software and computer effort for the EPIC collaboration.

EIC Software Statement of Principles

- As part of the "Lessons Learned" process, the entire EIC community came together to create a community document to define our aspirations for software and computing for the EIC
- Meant to form a sound foundation to design our software stack
- This document was spread to the entire EIC community through several rounds of open suggestions and endorsement to ensure this is truly a community document
  - Endorsed by a large group representing the international EIC community.
  - 100% of responses were positive!



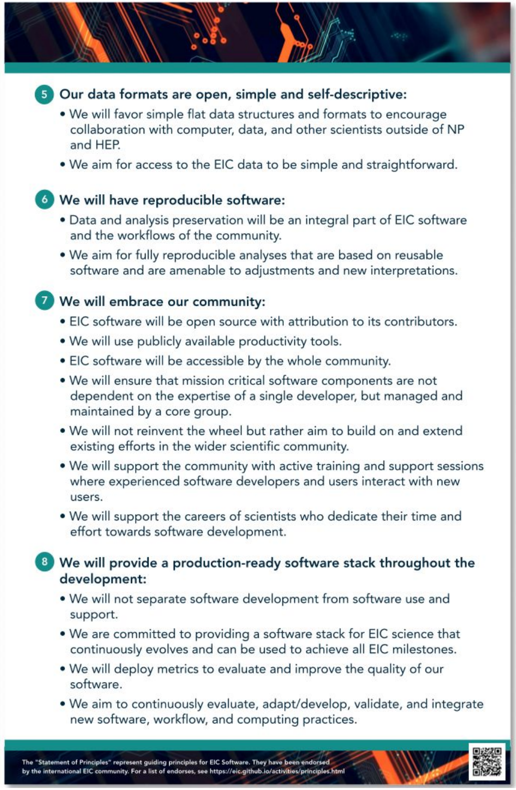
EIC SOFTWARE:
Statement of Principles

1. **We aim to develop a diverse workforce, while also cultivating an environment of equity and inclusivity as well as a culture of belonging.**

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.

The "Statement of Principles" represent guiding principles for EIC Software. They have been endorsed by the international EIC community. For a list of endorses, see https://eic.github.io/activities/principles.html

Endorsers from the international EIC community

W. Armstrong (Argonne National Laboratory), M. Asai (Jefferson Lab), J. Bernauer (Stony Brook University), A. Bressan (University of Trieste and INFN), G. Bozzi (University of Cagliari and INFN Cagliari), W. Deconinck (University of Manitoba), M. Diefenthaler (Jefferson Lab), C. Dilks (Duke University), D. Elia (INFN Bari), P. Elmer (Princeton University), C. Fanelli (Massachusetts Institute of Technology), S. Fazio (University of Calabria and INFN Cosenza), O. Hen (Massachusetts Institute of Technology), D. Higinbotham (Jefferson Lab), T. Horn (Catholic University of America), J. Huang (Brookhaven National Laboratory), A. Jentsch (Brookhaven National Laboratory), S. Joosten (Argonne National Laboratory), K. Kauder (Brookhaven National Laboratory), D. Keller (University of Virginia), J. Lajoie (Iowa State University), E. Lancon (Brookhaven National Laboratory), J. Landgraf (Brookhaven National Laboratory), P. Laycock (Brookhaven National Laboratory), D. Lawrence (Jefferson Lab), W. Li (Stony Brook University), J. Osborn (Oak Ridge National Laboratory), B. Page (Brookhaven National Laboratory), M. Potekhin (Brookhaven National Laboratory), A. Puckett (University of Connecticut), J. Reinhold (Florida International University), J. Rittenhouse West (Lawrence Berkeley National Laboratory), D. Romanov (Jefferson Lab), T. Sakaguchi (Brookhaven National Laboratory), B. Sawatzky (Jefferson Lab), A. Schmidt (George Washington University), R. Singh (Institute of Nuclear Physics Polish Academy of Sciences), P. Steinberg (Brookhaven National Laboratory), Z. Tu (Brookhaven National Laboratory), T. Wenaus (Brookhaven National Laboratory).

EIC SOFTWARE:
Statement of Principles

EPIC

1. We aim to develop a diverse workforce, while also cultivating an environment of equity and inclusivity as well as a culture of belonging.

EIC SOFTWARE:
Statement of Principles

EPIC

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.

Provisions for streaming readout from the start

Software design should not limit what systems we can run on, including future HTC and HPC facilities

Our design should be resilient against changing requirements, which we can accomplish by building a toolkit of orthogonal components.

EIC SOFTWARE:
Statement of Principles

EPIC

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.

Users should not need to know the entire toolchain to make meaningful contributions to a single component. Modularity helps here too.

We will make it easy for people to get started, and will avoid unnecessary requirements

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.

Simple, flat data structures will lower the bar for entry for new users, make it easier to accomplish data and analysis preservation, and facilitate multidisciplinary collaborations, e.g. with data scientists.

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.

Data and analysis preservation is a hard problem, rarely effectively addressed. We will consider this from the start. This also includes reproducible software.

EIC SOFTWARE:
Statement of Principles

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.

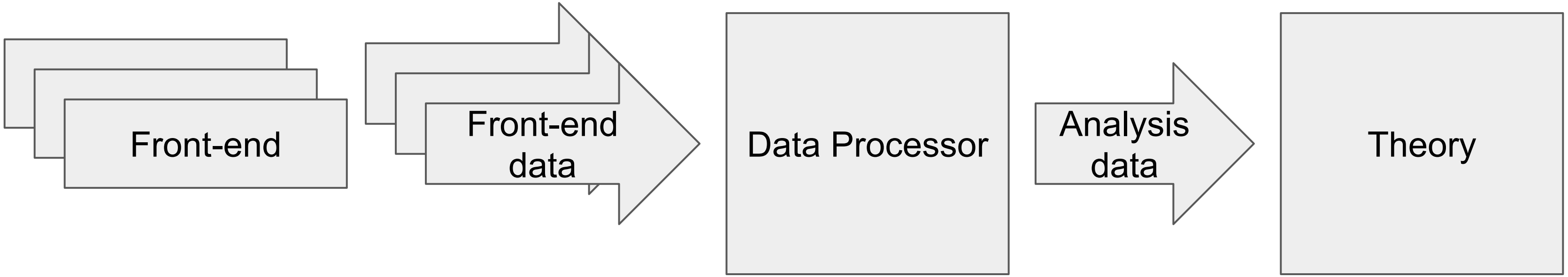
We will use existing community tools where possible and sustainable rather than reinventing the wheel. This allows us to focus less collaboration resources on framework tasks, and more on actual content (reconstruction, geometries, …)

We believe in an open source model, which has a track-record of success in particle physics. Additionally, open development will automatically help with career support of scientists that dedicate time on software.

We have deliverables for each of the CD milestones. We will ensure our new development goes hand-in-hand with continuous reliability to ensure the EIC detector and its science program are successful. Our modular approach will facilitate controlled and reproducible incrementalism.

EPIC

Streaming readout for the EIC



- Integration of DAQ, analysis, and theory will optimize the physics reach for the EIC
- Aim for a research model with seamless processing from sensor through DAQ to analysis and theory.
- Need to consider this from the start to ensure we build the best detector that supports streaming readout and fast algorithms for alignment, calibration, and reconstruction in real time or near real time
- Streaming readout and AI work hand-in-hand to enable a rapid turnaround from data taking to physics analysis and publication.

The need for a unified EPIC software strategy

- The proposal period saw a fragmented approach including different major frameworks and many smaller standalone projects.
- We need to unify our efforts to make the EIC detector a success, starting today throughout all CD milestones and into operation.
- We strongly believe in the EIC Software Statement of Principles, an effort of the entire EIC community under the umbrella of the EICUG.
- We will embrace these practices today to avoid starting our journey to EIC with technical debt.
- **We are writing software for the future, not the lowest common denominator of the past!**

Who are we?

EPIC CompSW
Software and Computing Conveners

Andrea Bressan
INFN
andrea.bressan@ts.infn.it

Cristiano Fanelli
William & Mary
cfanelli@wm.edu

Sylvester Joosten
ANL
sjoosten@anl.gov

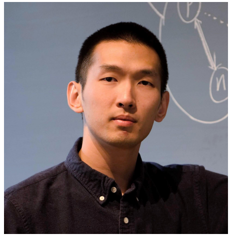
David Lawrence
JLab
davidl@jlab.org

EPIC SimQA
Simulation, Production, and Quality Assurance Conveners

Wouter Deconinck
U. Manitoba
wouter.deconinck@umanitoba.ca

Wenliang (Bill) Li
Stony Brook
wenliang.li@stonybrook.edu

Joe Osborn
BNL
josborn1@bnl.gov



Zhoudunming (Kong) Tu
BNL
zhoudunming@bnl.gov

Who are we?

EICUG SWG Conveners

Mailing list (135 members): eicug-software@eicug.org

Meetings (22 meetings): https://indico.bnl.gov/category/301/

How do we get there?

A critical path towards a unified software approach for the EIC

1. Assessment on the software solutions (pro & con list) together with the SimQA and DAQ working groups, guided by the EIC Software Statement of Principles.
2. Propose conclusion and recommendation to collaboration management and Project **by the Summer EICUG meeting**.
3. Software choice treated as any other technology choice? Independent review in the Summer**.**
4. Once decision is made, all new development should go in the official framework.
5. **Aim to have fully transitioned to the official software by October.**

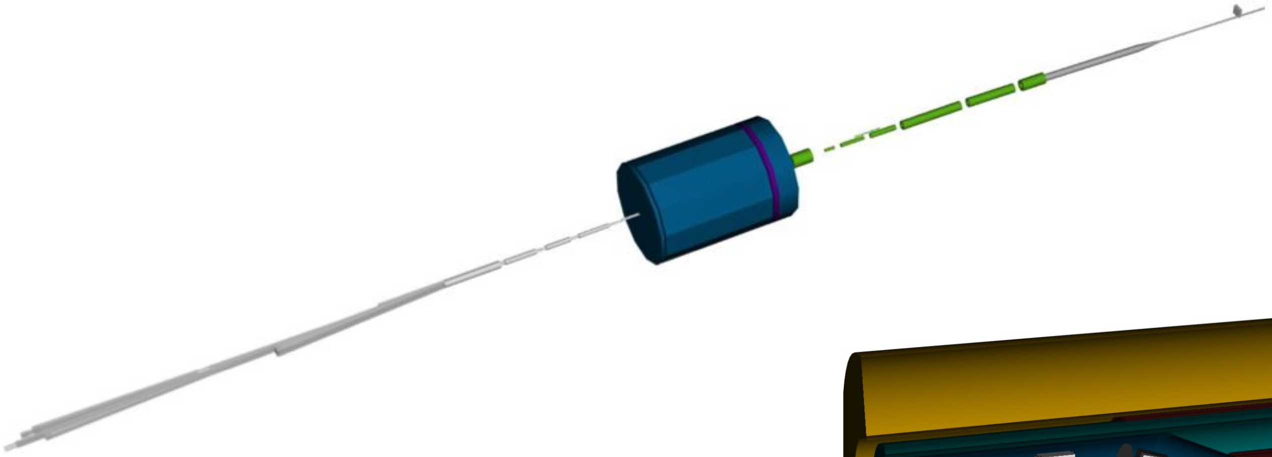
Procedure to Travel the Critical Path

1. Publicize schedule of topics with dates of discussion and decision
2. Assign chair for each topic. Chair will be POC for the topic. Responsibilities are:
   a. Organize discussion session agenda
   b. Publish draft list of requirements for the software being discussed at least 1 week in advance.
   c. Form list with at least one choice for the software to adopt to address the topic
   d. Collect suggestions for modifications to the requirements list and/or the software choices list
   e. Lead discussion on topic, starting with requirements list and the list of options
3. Presentations may be made regarding a specific decision topic, but should be communicated to discussion lead in advance for purposes of scheduling.
4. Use guiding principles from the EIC Software Statement of Principles
5. Discussion is required for all topics (formal presentations only as necessary).
6. **Based on the meeting, the joint CompSW and SimQA WG conveners will propose a single option, which will be open for comments and endorsement for one week.**

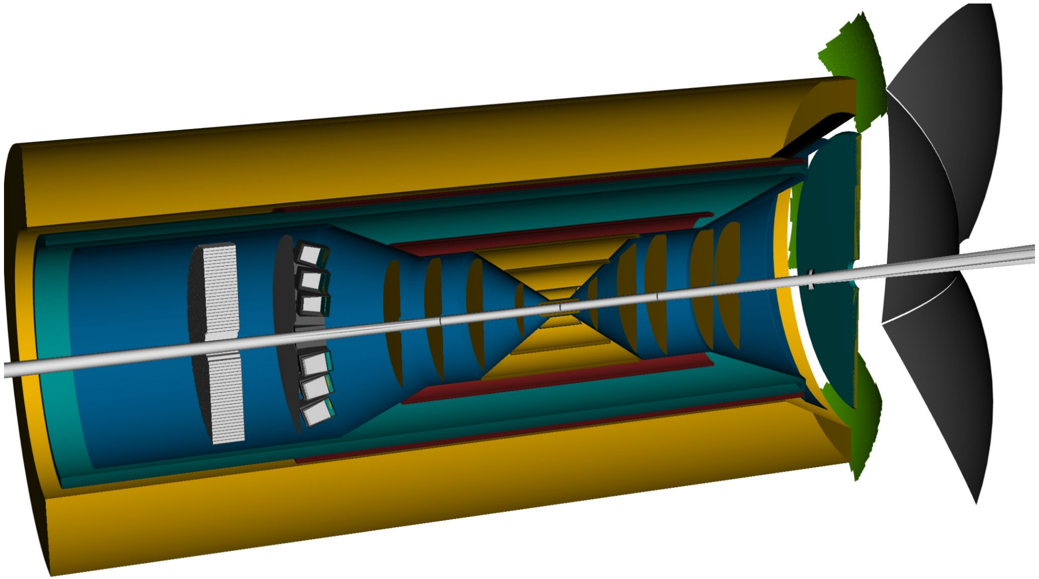
The biggest four decisions

		Discussion topic(s)	Decision topic(s)	*comments*	Point of Contact
May	**4**	*AIWG*			
	11	Transition Period	Present procedure. Decide on list and order of decision topics		
	18	*No meeting (Streaming Readout X Workshop)*			
	25	Code Repository ✔	Repository: - Location (GitHub, GitLab+Host) - Admins - Access		David Lawrence
Jun	**1**	Discussion Schedule	Schedule: - Decide most critical decisions to make before July 27th EICUG meeting - Schedule of topic discussions		
	8	Geometry ✔	Geometry: - Package (e.g. DD4HEP)		Markus Diefenthaler
	15	Data Model ✔	Data format - Generated events - Simulated data - Processed data (e.g. ROOT w/ specific tree format)		Whitney Armstrong
	22	Data Model			
	29	Reconstruction Framework ✔	Reconstruction Framework - Package		Wouter Deconinck
Jul	**6**	Reconstruction Framework			
	13	Data and Analysis preservation 🚧	Data Preservation - What is preserved (simulated, DSTs, ...) - Location(s) - Access (S3, xrootd, rucio, ...)		Kolja Kauder
	20	Documentation 🚧	Documentation: - Location of User documentation (wiki, repository,...) - Who will set up skeleton with list of topics (e.g. "Getting Started")		Dmitry Romanov
	27	*EICUG Meeting*			
Aug	**3**	Schedule realignment/ WG Business			
	10	Overview reports	DD4hep, JANA2, CI, ...	*reports*	

Geometry description and detector interface



Markus Diefenthaler
JLab
mdiefent@jlab.org



We picked **DD4hep** as tool to describe the detector geometry and to provide the detector interface for the reconstruction algorithms.

The entire EPIC geometry is already implemented in DD4hep.

Data Model

We will leverage the existing projects **`podio`** and **`EDM4hep`** to provide a standardized flat data model, accessible to researchers with modern AI/ML tools, on a variety of hardware and software systems.

For those aspects that are not in `EDM4hep` due to scope considerations, we will extend the data model with our own data definitions. We have experience with this from the proposal stage.

The standard data model for EIC will allow modularity and experimentation with new methodologies for data analysis.

Wouter Deconinck
U. Manitoba
wouter.deconinck@umanitoba.ca

EPIC

Reconstruction Framework

David Lawrence
JLab
davidl@jlab.org

- We selected **JANA2 as the reconstruction framework** based on a carefully formed set of requirements reviewed by the EIC software community
- Development of the JANA2-based EICrecon software is well underway starting with porting of the relevant algorithms already developed for EIC detector proposals, on-schedule to meet the October deadline.

EPIC

Data and Analysis Preservation

Kolja Kauder
BNL (NPPS EIC Liaison)
kkauder@bnl.gov

"[A]spects to consider beyond a choice of software [...] include policy decisions that will require endorsement from the collaboration as a whole and resources to back them up. A **task force assigned to this purpose** was called for in the discussion. [...] The task force will be organized by two interim co-leads until the official formation of a collaboration."

Production Strategies

Joe Osborn
BNL
josborn1@bnl.gov

- Following the software decision schedule, the distributed workflow management system discussion to be held in forthcoming weeks
- Technical solutions deployed by both proto-collaborations in proposal stage are **not** adequate long term
- EIC S&C community has engaged with development teams of available technologies, e.g. DIRAC and PanDA

Code repository and Continuous Integration

The software infrastructure will use a **hybrid solution** that combines the benefits of public and accessible **code repositories on GitHub** with powerful and scalable backends with **self-hosted GitLab servers for continuous integration**.

The transition of repositories and software projects from GitLab servers used in the past is well underway, and proceeding on schedule towards completion by October.

Implementation of integration of GitHub continuous integration with self-hosted GitLab servers is proceeding ahead of schedule.

Wouter Deconinck
U. Manitoba
wouter.deconinck@umanitoba.ca

EPIC

Leveraging AI and ML from day 0

Cristiano Fanelli
William & Mary
cfanelli@wm.edu

- The recently formed EPIC collaboration is quite active in AI/ML, and as a matter of fact EPIC can be one of the first experiments to be designed with the support of AI
  - The number of AI/ML activities is anticipated to grow in the next few months
- Lots of work has been recently done to determine the SW stack for the collaboration (DD4Hep, data model, JANA2), a fundamental step towards the CD2/3a .
- From an AI/ML perspective, several of these features seem forward-looking and allow for AI/ML applications and utilization of heterogeneous resources.
- Large-scale AI/ML applications entails specific infrastructure needs that require additional discussion
  - ML lifecycle, distributed training, etc
- The EIC community is engaged in AI/ML activities, and the AI4EIC WG is a good forum to address these important aspects. More info on meetings and workshop in https://eic.ai/events

EPIC

September will be our training month

We will have September training sessions on **Thursday afternoon 2:30pm to 5pm Eastern** and **Friday morning 8am through 10:30am ET** to serve all collaborators worldwide.

- *September 1/2*: Setting up your environment for collaborative EPIC development
- *September 8/9*: Geometry development using DD4hep: how to modify or add detector description?
- *September 15/16*: Simulation of single particles or physics events using geant4 and ddsim
- *September 22/23*: Reconstruction algorithms in JANA2: from geant4 output to reconstructed quantities
- *September 29/30*: Writing physics benchmarks that run automatically and reproducibly

Software for *all* EIC experiments - and beyond

- Nothing about the modular software toolkit design is unique to Detector 1
- We explicitly expect the toolkit to be used as a starting point for the Detector 2 software toolkit
- Many design decisions were taken to explicitly allow collaboration and even algorithm sharing with other NP and HEP experiments
- The EPIC software stack could be used for future NP experiments, e.g. SoLID at Jefferson Lab (a fixed-target experiment!)

Component	**Modification for detector 2?**
Geometry	New configuration, can reuse many detector components
Data model	Identical
Framework	Can reuse/add to algorithms, only need different configuration
Code repository and CI	Same resources could be used
Data analysis and preservation	Same strategy can apply
AI and ML	Same strategy can apply

Bottom line: detector 2 can hit the ground running!

Summary

- Journey to our unified EPIC software stack well underway.
- All major topics have been addressed!
- This review is the next major milestone.
- September will be training month, and full transition of software stack will happen in time for second large simulation campaign in October