Validation WG – Status and Plans

Torri Jeske, Dmitry Kalinkin
April 22, 2024

The charge

Charge:

- Responsible for the validation of the simulations via a suite of detector and physics performance plots
- Develop autonomous checks and verification (CI)

Priorities for FY23-24

- Implement and document our Simulation Production Strategy, together with the Production WG
- Develop and maintain a collection of plots that showcase the performance of the ePIC detector, its physics reach, and enable comparison to a baseline or previous simulation campaigns.
- Drive the development of unit tests for the ePIC software, together with the Development WGs
- Design and implement a Software and Analysis Validation Policy containing enforceable testing requirements in line with our Statement of Principles

Dmitry: The last bullet point received least amount of attention. There wasn’t a clear endorsement of the VWG’s benchmark framework to be used for production of TDR-related figures. We can still continue the discussion as part of the Publication Policy review and the surrounding Data Analysis and Preservation discourse.

ePIC year 2023-2024 Highlights

Please summarize the first year’s highlights on another slide.

- Partial adoption of Snakemake-based workflows showed that the software, indeed, meets our current needs
- Lots of improvements for development support by benchmarking
  - More tests running as GitHub actions for epic and EICrecon
  - Full container build is now triggered for eicweb EICrecon
  - Capybara reports are well integrated into the PR review process
- Initial tutorial on benchmark development https://github.com/eic/tutorial-developing-benchmarks
- New detector benchmarks: “backgrounds”, “ecal_gaps”, “material_scan”
- New physics benchmarks: “diffractive_vm”

Current priorities

- Extending the pool of useful benchmarks
  - Getting things battle-tested for wider use by the collaboration
- Develop tooling for performing comparisons
  - This originally focused on image artifacts produced in CI system, but this came with a lot of issues
    - building custom <img> tags per image
    - not all image artifacts are generated all the time

Collaboration engagement

Our current workforce is limited to just few people in the Operations WG (~3 people). There were no WG meetings.

It would appear that most of the existing initiative drowns in procrastination, it is hard to get people to submit less-than-ideal code.

Our private outreach is usually met very positively.

Examples of tasks

Examples of tasks for collaboration members interested in becoming involved

- Contributing detector studies as detector benchmarks
- Contributing physics analyses as physics benchmarks
- Porting existing benchmarks to Snakemake
- Developing unit tests for EICrecon algorithms
- Upstream fixes for issues with Snakemake 8 (to upgrade from version 7)
  - https://github.com/snakemake/snakemake-storage-plugin-s3/issues/24
  - https://github.com/snakemake/snakemake-storage-plugin-xrootd/issues/2