Contribution ID: 14

JADE v4, a more robust and expandable architecture for neutronics V&V

Thursday 15 May 2025 10:00 (30 minutes)

In the last couple of years, JADE development has been pushed forward by Fusion For Energy and UKAEA with a new objective: use the JADE framework also to perform code to code comparisons. This new collaboration led to JADE v3, a first proof of concept that ported the tool on the linux platform and allowed for the first time to run benchmarks not only with MCNP but also with OpenMC. During this phase, the JADE environment was restructured and components with different scopes were made independent and separated in different repositories. Despite all these achievements, it became clearer that adding new features to JADE was becoming increasingly harder because JADE was not initially conceived with this broader scope in mind. The code base reached a high degree of complexity and was becoming a barrier for the onboarding of new developers in the project. These reasons led to the work presented here, which focus on the latest iteration of JADE development: JADE v4. The entire core architecture has been refactored following two main principles. The first is that JADE is conceived now from the very beginning to be a framework for comparison of code-library results and not simply for library to library or for code to code. The second is that the csv data produced by JADE, which are essentially the results of the different simulations in a table format, will be now a key interface. That is, all post-processing that is transport code dependent will end with the production of the csv. This allows the JADE post-processing (plots and excel summaries) to be completely transport code independent. A side benefit of this is that a better interface is created for the JADE web app. Moreover, this will allow to expand the JADE benchmark suite only through configuration files and input templates, without the need for additional coding. Finally, the extensive refactoring also allowed to implement a series of software design best practices which significantly increased JADE robustness and expandability.

The following is a list of the main achievements reached by JADE v4:

- Re-think JADE to be a code-lib 2 code-lib comparison. Not code2code or lib2lib.
- Full implementation of OpenMC (for the available benchmark inputs)
- Implement a structure that clearly isolates (by inheritance of specific abstract classes) the extra python code needed to support an additional transport code.
- No additional programming for new benchmarks (including experimental). Everything is handled by ad hoc YAML configuration files.
- Raw data (csv files) are now a stronger interface. They are the end point of transport code-dependent processing and the starting point of the JADE plotting and excel. This facilitates the integration with 3rd party post-processors like the WebApp.
- Software design best practices were applied during refactoring.
- The codebase (number of lines) has been reduced to only 1/3 with respect to v3.1.0 one.

Author: LAGHI, Davide

Presenter: LAGHI, Davide

Session Classification: Information Technology, Tools, Codes