Synergistic Opportunities Between
Data Management Tools and Integrated Modeling Frameworks

* O. Meneghini (GA), S. Smith (GA)
* Corresponding Author: meneghini@fusion.gat.com

Request Oral Presentation: Yes
Primary Panel: F **Secondary Panel:** G

Summary: This white paper argues for the need for rich data management tools for both simulation and experimental data analysis. The integration of these tools at the framework level will have a significantly broader and more rapid impact on our scientific research.

Data management has not been a significant consideration in the development of most existing FES simulation software. Current practice in fusion simulation data management primarily uses file-naming conventions, applied on a per-code and/or per-user basis. As simulations use more model interactions and produce larger volumes of data, a growing emphasis will need to be placed on maintaining data integrity within a simulation, and guaranteeing long term, postmortem usability of data artifacts that simulations produce. It is not the mere existence of data that is important, but our ability to make use of it. Generous provisioning of metadata, including data provenance and data relations, are needed to allow traceability of results, extend data shelf life, and enhance scientists' ability to find and share information. Data management initiatives such as the Metadata Provenance Ontology [MPO] project tackle these issues by providing the flexible APIs and back-end infrastructure that are needed to support specification, collection, and inspection of data analysis and simulation workflows.

Although such data management tools provide a complete solution to answer data lifecycle questions, their adoption is likely to be hampered by the significant initial investment cost for the perceived additional value to scientists' research. In fact, proper end-to-end data management would require code developers to instrument their tools, scripts, and codes. Similarly, users have to keep track of what data goes where as they proceed in their research. Doing this without breaking the thread can be a challenging task, especially considering that the path of successful research typically has plenty of forks, crossings, and dead-ends. Also, one has to consider that for the most commonly cited application examples the stakeholders are not the users/developers that are charged with the burden of implementing these code changes, and tracking the data. A notable example is the implementation of these features as a way to satisfy the DOE Digital Data Management Statement [DoE July 2014 – All stages of the digital data lifecycle: capture, analysis, sharing, and preservation].

It is likely that without lowering the upfront cost and increasing the users' perceived value for data management, the widespread adoption of these tools will be very difficult. Integrated modeling frameworks could offer the solution to these issues. First, the frameworks can be instrumented for data management and tracking purposes. Second, they can provide users with a unified interface for handling both experimental and simulated data, as well as create, manage, and execute research workflows thereby accelerating their research.

We propose to reduce the user investment cost by instrumenting, on their behalf, the workflow engine and the interactive environment of integrated modeling frameworks with data management instructions. By implementing the data management at the framework level, the information regarding code workflows, as well as their inputs/outputs, storage location, execution time, execution server, simulation requirements, and outcome can be automatically tracked. For example, users of the OMFIT [gafusion.github.io/OMFIT-source] integrated modeling framework can perform their experimental analyses and predictive simulations all from within the framework. By integrating the MPO tracking capabilities within the OMFIT framework, these users will not have to do anything other than using the framework as they regularly do to have their data managed (and take advantage of all of the associated benefits, including satisfying the stringent DoE Digital Data Management requirements).

Integration at the framework level offers the benefit of providing a minimum level of information tracking, upon which searches and statistics can be made. Also, all workflows would be tracked (though users should still be given the possibility to opt-out from such tracking). The availability of a larger pool of users from the integrated modeling framework should naturally increase the value of the database, since some questions that the data management system can answer are only interesting with a community/activity of a minimum size: “Who does analysis X so I can ask for advice?”, “Who else is analyzing this shot in detail?”, “Who are the users of my code?”, “How is my code being used?”.

Users will be able to provide supplemental metadata information (such as comments, simulation quality metrics, etc.) within their workflows. This should be easier to do than within the physics codes, since workflows are usually programmed in an interpreted language (e.g. Python) rather than being compiled. Significant value could be added to the data management system when workflows will be instrumented to track information regarding code inputs and outputs, since this will enable scientists to discover patterns and create reduced models based on aggregated sets of experimental analyses or first-principles simulations.

A limitation of this approach is that any tracking of information at the granularity of within a code will not be captured. For example, the tracking of paths through a code’s choices at logical forks during execution will be missed. But most likely in the vast majority of cases, such detail is not required. In such cases where a developer feels that it is needed, then the individual code would need to be instrumented. Therefore, it seems to follow that whatever infrastructure is adopted to provide data management at the framework level would also work at the code level. In other words, the methodology at the framework level or code level is the same, but the benefit at the framework level is greater.

Integrated modeling frameworks will also benefit from the additional value that data management tools can provide, since scientists will see adopting such frameworks as a way to improve their research and simultaneously receive the benefit of effective data management and tracking. As a result, we expect that the synergy between data management tools and integrated modeling frameworks will result in improved user adoption of both FES technologies.