

NATIONAL IGNITION FACILITY (NIF) EXPERIMENT INTERFACE CONSOLIDATION AND SIMPLIFICATION TO SUPPORT OPERATIONAL USER FACILITY

A. Casey, E. Bond, B. Conrad, M. Hutton, P. Reisdorf, S. Reisdorf.
Lawrence Livermore National Laboratory, Livermore, CA 94550 USA

Abstract

The National Ignition Facility (NIF) at the Lawrence Livermore National Laboratory is a 192-beam 1.8 MJ ultraviolet laser system designed to support high-energy-density science. NIF can create extreme states of matter, including temperatures of 100 million degrees and pressures that exceed 100 billion times Earth’s atmosphere. At these temperatures and pressures, scientists explore the physics of planetary interiors, supernovae and thermonuclear burn. In the past year, NIF has transitioned to an operational facility and significant focus has been placed on how the users interact with the tools necessary to conduct an experiment at NIF. The current toolset was developed with a view to commissioning the NIF and thus allows flexibility that most users do not require. The goals of this effort include enhancing NIF’s external website presence, easier proposal entry for NIF experiments, reducing both the amount and frequency of data the users have to enter, and simplifying user interactions with the tools while reducing the reliance on custom software. This paper will discuss the strategies adopted to meet the goals, highlight some of the user tool improvements that have been implemented and planned future directions for the toolset.

INTRODUCTION

In the past year, the National Ignition Facility (NIF) [1] has transitioned to an operational User Facility and the tools accessed by users needed to be updated accordingly. The current tools were designed to support facility commissioning and have the following characteristics:

- Designed for commissioning a specific function and are integrated with other tools after the fact
- Provide maximum flexibility and options over simplicity
- Tool access is assumed to be from on-site
- Users assumed to have expert level knowledge and experience with the tools

To get an idea of what this looks like to the user, Figure 1 lists the tools that are be used to define an experimental setup on NIF. Even for the expert user, this is a significant number of tools and hence options that the user has to understand. This ultimately detracts from the real task of defining and executing experiments.



Figure 1: Users Need to Interact with Several Experiment Setup Tools at NIF

The goal of the team was to make interacting with NIF analogous to buying an airplane ticket, Figure 2. The user of the airline website needs to only know some high level details of what they want to accomplish such as departure and return dates, number of people in the party etc. They do not need to know about plane weight, fuel load balances, airport operating hours etc. that are needed by the airlines in order to operate safely and get the customer to their chosen destination. In the same way, to conduct an experiment, the user should only need to provide the information relevant to the task that they are trying to complete.

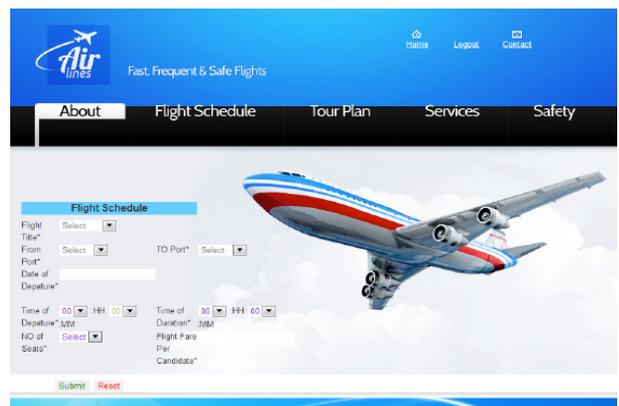


Figure 2: A web site that supports simple user interaction.

DEFINING THE PROBLEM SCOPE

Solving all of the usability issues in a single update was considered to be too risky and too difficult to implement whilst the tools were being used to support production. So, very early in the process, it was decided that the problem should be broken into three phases.

- Phase one would focus on the start of the experiment lifecycle where the users respond to the request for proposals.
- Phase two would implement a common data framework supporting integration of all of the tools as well as a simplified experiment setup
- Phase three would flow this data through the rest of the processes of experiment planning, experiment definition, facility configuration and ultimately, execution and data analysis.

For soliciting proposals, the stakeholders wanted an external website that could be used without the formality of a login plus additional credentials. It would also allow for collaboration between users; be a source of initial experiment setup that could be used by other tools later in the lifecycle and would have the capability to expand to offer other features such as document configuration, experiment status information and facility metrics to name but a few.

To achieve all of these goals, the team needed to consider a number of different criteria (Figure 3) that could have significantly impacted how they were to proceed. Being on the outside of the lab firewall had significant implications for the IT infrastructure in terms of the cost of procuring the hardware; responsibility for maintenance and patching; as well as monitoring of user events such as administrators logging in, updating of records etc.

The team could have built a custom tool from scratch but there would be worries about security and increasing the amount of code that needed maintenance. Not only

that, but the solicitation of proposals is really a customer relationship management (CRM) problem and there were many vendors in the market providing solutions to this kind of process. That said, other tools that the team looked at were either highly integrated into the tool developers own process (as at other national labs for example) or tended to focus more on a sales model that had a look and feel which did not fit with our scientific mission.

Our final consideration was integration with our current tools and the skills that our own developers had. Writing the application ourselves would not present a problem but using a commercial off the shelf (COTS) product might. There would need to be a rich API to access the data and it would need to be compatible with Oracle database technologies that are widely used at NIF.

SELECTING A PARTNER

After much deliberation, the decision was made that the team needed a partner to help create an external website with all of the security requirements that that entails. The team examined Amazon Web Services [2], Google Cloud Platform [3], Microsoft Azure [4], Salesforce [5] and hosting a web page based on Oracle Apex [6] to name but a few. In most cases, infrastructure-as-a-service (IaaS) would require the team to develop all of the software for the application on that infrastructure. Oracle Apex could provide a more complete software solution but the team would need to maintain the hardware and infrastructure externally.

In the end the team decided to go with a platform-as-a-service (PaaS) approach and selected the Salesforce platform. The selection was based on the fact that that CRM platform came closest to meeting all of our critical requirements which are:

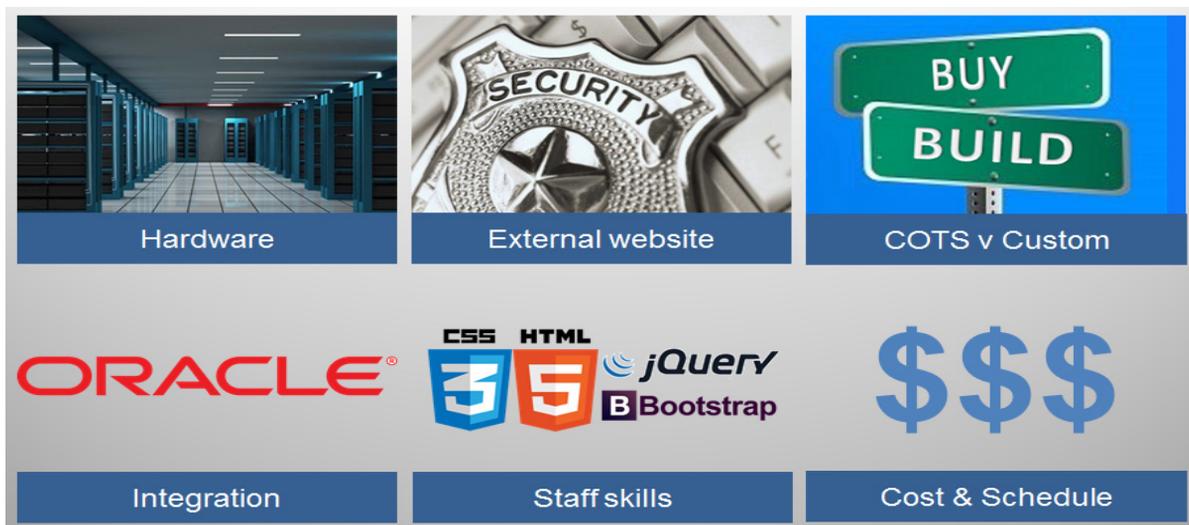


Figure 3: Considerations for selecting a platform for developing an external website.



Figure 4: With all tools there are advantages and disadvantages, Salesforce is no different.

- Hardware maintained by vendor
- Security handled by vendor
- Oracle based like current NIF tools
- Rich API integrates with commonly used technologies; JS, HTML5 etc.
- Minimized development time and effort
- Ability to customize

EXPERIENCE WITH THE SALESFORCE PLATFORM

As this was the first experience with Salesforce, the team decided to use consultants in order to develop the design. The primary goal was to ensure that they followed best practices in terms of Salesforce design and the use of the available components and most importantly, that the team understood the user requirements sufficiently that they could be described to a partner unfamiliar with the problem.

Working with NIF Computer Security was critical in getting the application deployed. Fully understanding their needs and concerns was more difficult than initially anticipated. Despite initiating contact with them well in advance of the projected delivery date, assembling the necessary paperwork and getting an approval to operate (ATO) came close to the delivery date.

Building an application on a platform provided by a vendor always has its strengths and weaknesses and that is also true with the Salesforce platform (Figure 4).

Implementing the proposal tool, the team was required to develop a data model that allowed for users to have different roles depending on the proposal they were working on. They could be a principle investigator (PI) and a collaborator in one call and be a reviewer in another. Fortunately, the platform has very strong role management built into it. Utilizing user profiles and object level security, the team was able to develop the

necessary access logic within the framework on the platform.

A proposal tool is essentially a process and workflow engine; requesting proposals, handling the user responses, and then providing notifications to the user of their status. Using triggers and processes, the platform supports automating manual activities such as sending email when a value in an object changes.

Unit testing is a vulnerable part of the development lifecycle as, when deadlines approach, it is often reduced in scope in order to meet the schedule. In order to promote code from a development sandbox into production, Salesforce requires that the module has at least 75% source code coverage. This does not guarantee the quality of the testing being performed, but it does set a minimum expectation from the outset of development that cannot be ignored.

From the beginning of the project, users wanted a way to communicate and collaborate. The Salesforce platform has its own tool Chatter for this purpose. It interacts with email and also provides a history of your communication with an individual or group. The inclusion of this feature meant that the team did not have to integrate an existing tool or write a custom app which was very much in accordance with the desire to minimize the development of new code as much as possible.

The biggest issue the team had with the platform was the native Salesforce development environment, and the lack of a debugger to help diagnose runtime problems. For integrated development environments (IDEs) such as Eclipse [7] and IntelliJ [8] there are plugins available that support the Salesforce platform and they go some way to providing modern development features such as code completion, source code validation, and integrated API documentation. At DreamForce '15 [9], Salesforce announced that their winter '16 platform release would include a debugger.

The last issue of note is that sometimes the developer is not able to do everything that they want because of the way the framework operates. For example, the developers found that in certain circumstances, they could only get the ID of a proposal object to display when the name of the proposal was desired. There were ways around these types of limitations, but it takes more work and research than might be expected.

THE NIF USER PORTAL & FUTURE WORK

In July 2015, the team launched the NIF User Portal with a single application, the proposal tool on the SalesForce platform (Figure 5), completing phase one of the planned updates. It has been used on a Discovery Science call for proposals and will be used on all future calls. The new tool supports the issuing of the call itself, the handling of the users' proposals, and the reviews of those proposals.

Now that the platform has been established, the team will be focusing on phase two of the upgrade, the creation of an experiment editor and a common data framework. These updates will reduce the amount of data initially required from the user and allow what the user does enter to flow into the rest of the NIF toolset beginning the process of simplifying setup.

CONCLUSION

Having developed an external website for NIF using the SalesForce platform, there are a number of recommendations that the team believes should be

followed by any organization thinking of a similar undertaking:

- Get expert help with the design to make sure that the application follows best practices.
- Limit the scope of the initial deployment to reduce risk and provide for a greater chance of success.
- Get computer security involved with the project as early as possible.
- Limit the amount of customization to reduce the amount of ongoing maintenance

ACKNOWLEDGEMENT

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344

REFERENCES

- [1] E. I. Moses, "Overview of the National Ignition Facility", Fusion Science and Technology 54 (2008).
- [2] AWS website, <https://aws.amazon.com>
- [3] Google Cloud, <https://cloud.google.com>
- [4] MS Azure website, <https://azure.microsoft.com>
- [5] Salesforce website, <http://www.salesforce.com>
- [6] Oracle Apex website, <https://apex.oracle.com>
- [7] Eclipse website, <https://eclipse.org>
- [8] IntelliJ website, <https://www.jetbrains.com/idea>
- [9] DreamForce'15 website, <http://www.salesforce.com/dreamforce/DF15>

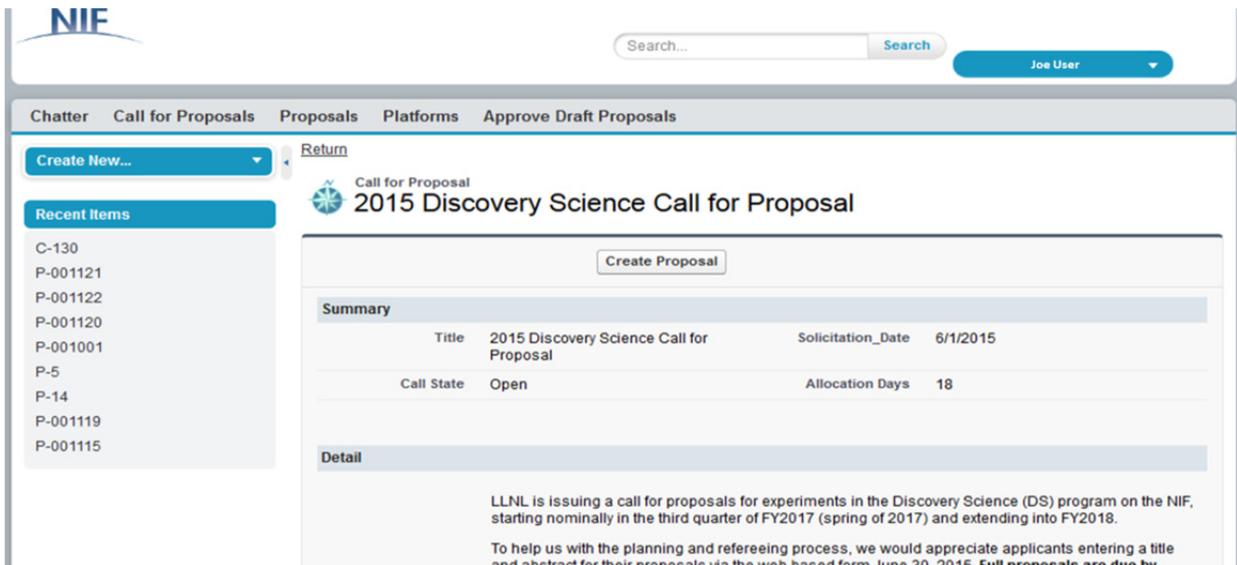


Figure 5: The NIF User Portal proposal management tool went live in July 2015. <http://nifportal.force.com/>