

## §26. Development of Web Interfaces to Analysis Codes

Emoto, M., Yoshida, M., Funaba, H., Nagayama, Y. (NIFS),  
Murakami, S. (Kyoto Univ.)

There are many analysis codes developed to study plasma physics of the LHD experiments. However they are difficult to use because 1) they run on super-computers, and they are command-line interfaces while most of the researchers use Windows or other GUI based operating system. 2) Co-researchers outside NIFS don't know where the data exist or how to retrieve them. Therefore, user-friendly and intuitive interfaces are required especially for visitors out from NIFS. In order to improve this situation, the author have been developing user friendly interfaces to the analysis codes, and the author chose web browser as its interface. This is because the web browser is one of the most common applications, and the user doesn't have to install programs. This feature is useful especially for the visitors out from NIFS because they don't have enough time to install the application to learn the complicated operations during their short term visits.

To demonstrate the usability of this approach, we chose to develop a web interface to FIT code [2]. FIT code is a simulation code to study energetic ion distribution in helical plasma developed by Murakami. The Figure 1 shows the overview of the web interface to FIT code. The system consists of three parts, client pc, web server, and super computers. The user uses an ordinal web browser to edit the parameters to submit jobs. The web server receives the request from the client and it relays the request to the supercomputer through the gateway servers.

In order to realize intuitive GUI, AJAX (Asynchronous JavaScript plus XML)[3] is used for the client application. AJAX is a technique to realize interactive web application using JavaScript. However, AJAX must use both JavaScript and HTML at the same time, and it worsens the readability of source codes and becomes hard to maintain the source codes. To solve this problem, the authors adopted Ruby on Rails [4] as an application framework. Because Ruby on Rails encapsulates HTML and JavaScript, the developer can use only Ruby to implement AJAX. Another advantage using Ruby is it become easier to deal with complex structured data. Because JavaScript doesn't provide standard function to exchange data between the client and servers, it is the developer's responsibility to encode and decode data, and it is difficult to handle the complex structured data. On the other hand, Ruby on Rails uses Ruby variable, it is easier to deal with them.

The supercomputer is isolated from the other LAN by the security reason. The user has to log in to the gateway server to submit jobs. The application server uses SSH to submit jobs and retrieves the results using SFTP. For now, the system uses a common user ID to submit jobs using SSH. The authentication method used here is public-key authentication. Therefore, it isn't impossible to use this account from other computer to log into the gateway server. However, it is susceptible to pranks to submit a lot of jobs

using this system. Therefore, the authors are planning to develop web based authentication.

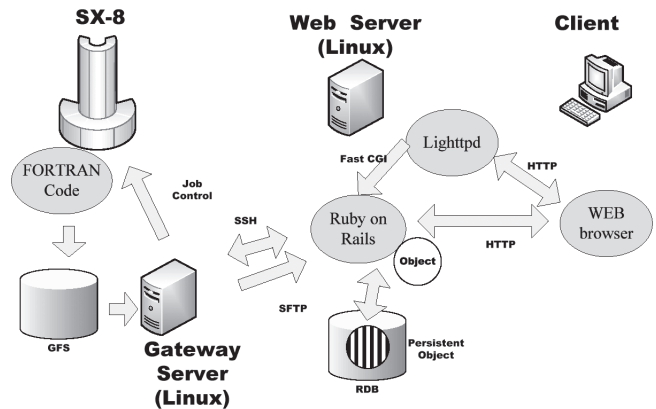


Fig.1 System Overview

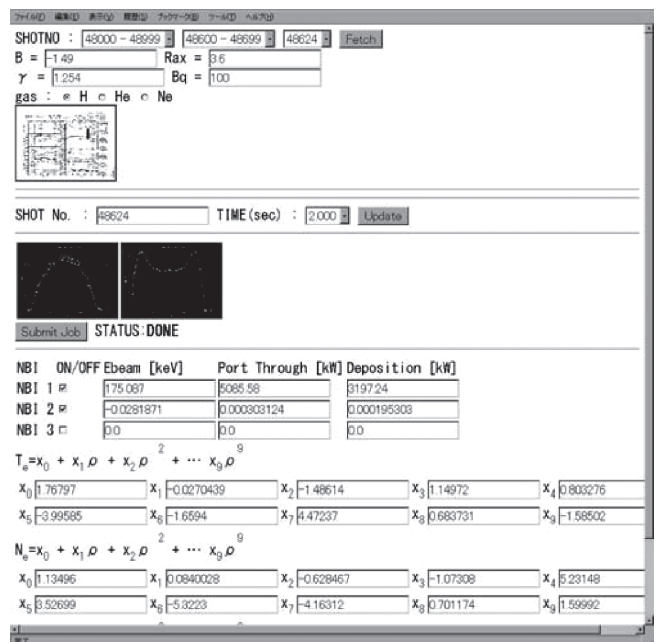


Fig 2. Web Interface to FIT code

### Reference

- 1) IAEA Technical Meeting, 2007, Inuyama, Japan
- 2) S. Murakami, et. al., Fus. Tech. 27 Suppl. S, (1995) 256-259
- 3) <http://www.adaptivepath.com/publications/essay/archives/000385.php>.
- 4) <http://www.rubyonrails.org/>