GLOBAL SEARCH TOOL FOR THE ADVANCED PHOTON SOURCE INTEGRATED RELATIONAL MODEL OF INSTALLED SYSTEMS (IRMIS) DATABASE *

D. E. R. Quock#, ANL, Argonne, IL 60439, U.S.A.,
M. B. Cianciarulo, Purdue University, West Lafayette, IN 46324, U.S.A.

Abstract

The Integrated Relational Model of Installed Systems (IRMIS) is a relational database tool that has been implemented at the Advanced Photon Source to maintain an updated account of approximately 600 control system software applications, 400,000 process variables, and 30,000 control system hardware components. To effectively display this large amount of control system information to operators and engineers, IRMIS was initially built with nine Web-based viewers: Applications Organizing Index, IOC, PLC, Component Type, Installed Components, Network, Controls Spares, Process Variables, and Cables. However, since each viewer is designed to provide details from only one major category of the control system, the necessity for a one-stop global search tool for the entire database became apparent. The user requirements for extremely fast database search time and ease of navigation through search results led to the choice of Asynchronous JavaScript and XML (AJAX) technology in the implementation of the IRMIS global search tool. Unique features of the global search tool include a two-tier level of displayed search results, and a database data integrity validation and reporting mechanism.

HISTORY OF IRMIS

Don Dohan, while working at the Argonne National Laboratory (ANL) Advanced Photon Source (APS), set to the task of developing a software framework for modeling the EPICS installed control software. He used a relational database as the backbone for describing how EPICS process variable attributes are linked to each other [1]. Subsequently, Ned Arnold, of ANL, and Don partnered together to expand the database schema to include installed control system hardware in a connection-oriented approach [2]. This relational database application became what is now known as IRMIS, and implementations spawned from IRMIS are currently in use at GANIL, the Canadian Light Source, TRIUMF, SLAC, BESSY, SNS, and PSI. Over time, and with the addition of more IRMIS developers, the IRMIS Web-based user interface at APS evolved to include nine separate "viewers" for displaying categorized, detailed information about control system hardware and software (see Figure 1).

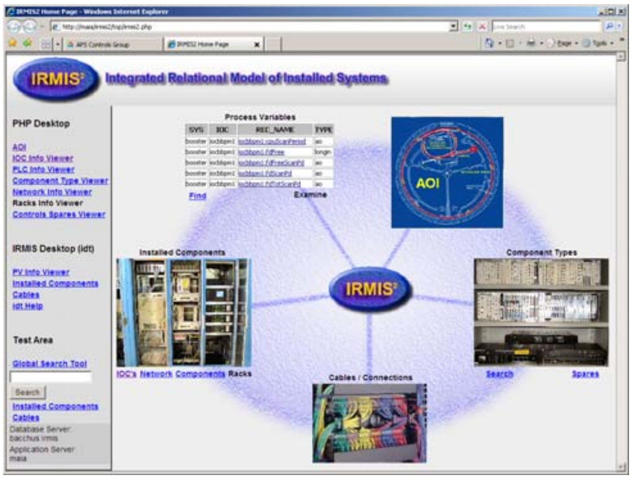

Figure 1: APS IRMIS home page.

GLOBAL SEARCH TOOL

A Global Search Tool (GST) for locating a string of text in any category of IRMIS data has recently been added to the suite of IRMIS viewers. AJAX technology was selected for building a highly responsive Web-based user interface. The motivation behind creating this Google-like keyword search tool for IRMIS is as follows:

- The IRMIS relational database at APS comprises seventy tables. An organized and logical means for quickly searching through this expansive amount of information when only a small amount of information is initially known by the user, such as a single word or abbreviation, was highly desired to aid engineers in rapidly troubleshooting real-time control system issues.
- IRMIS has six viewers written in PHP language. This type of Web page display requires complete reloading each time new data is retrieved from the IRMIS database. This makes for a sluggish user interface environment.
- The other three viewers are written in Java with Hibernate object relational database mapping software. The six PHP and three Java viewers are currently not connected to each other through software links, and for the most part, the PHP viewers are not interlinked – the user must use the browser back button and start over on their search of IRMIS data to get to other categories of control system software or hardware.
- The Java/Hibernate viewers are relatively slow when downloading the initial set of values retrieved from the IRMIS database, and once loaded, the

*Work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-06CH11357.

#quock@aps.anl.gov

information is static during the entire viewing session.

Software Architecture

The Global Search Tool is built with AJAX technology for the Web-based user interface, PHP for the server side processing of user requests, and MySQL as the IRMIS relational database. AJAX is not a specific software language, but rather a software design concept on how to combine JavaScript, HTML, Cascading Style Sheets (CSS), and Extensible Markup Language (XML) to make asynchronous calls to the server via the JavaScript XMLHttpRequest object. The XMLHttpRequest object is the key component of AJAX that provides the ability to refresh a subsection of a Web page with new data retrieved from a server while the remainder of the Web page remains intact and available for other user interactions. XML is the markup tag syntax used for transferring data from/to JavaScript running on the client (user's Web page) to/from PHP running on the Web server.

JavaScript ←→ XML ←→ PHP ←→ MySQL

Many excellent books for the novice have been published on AJAX [3].

User Interface

The Global Search Tool provides three levels of detail for defining the user's search criteria. All of IRMIS can be searched simply by typing in a string of text. One or more major categories of IRMIS can be selected for narrowing the search. The major categories currently defined include AOI, IOC, PLC, Component Type, Installed Components, and Process Variables. A user's search can optionally be further refined by selecting only from the minor categories of IRMIS. Each major category is broken down into its constituent minor categories as shown in Figure 2. For example, the major category IOC has minor categories: IOC Name, System, Location, Cognizant Developer, and Cognizant Technician.

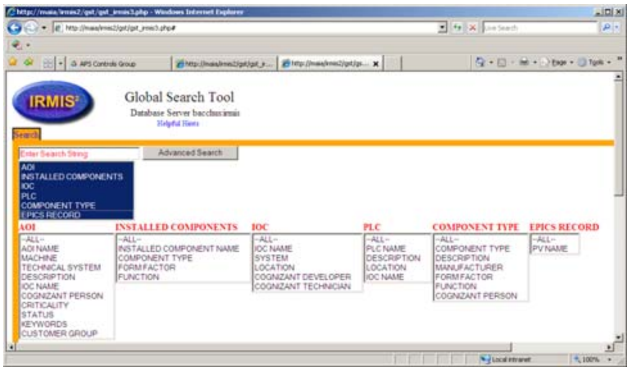

Figure 2: GST major and minor search categories.

The asynchronous behavior of the Global Search Tool is designed such that a new search of the IRMIS database is performed in response to every user keystroke in the text entry box. A minimum of three characters is required in the search string before a request is sent from the client to the server. A new search is not requested by the client in the case where: the current search string is over three characters in length and the current search resulted in zero matches -- the user then shortens the search string by deleting characters. This logic is used to not waste time processing a non-productive search request.

The search results are presented in two levels of detail. The first level of detail provided to the user shows the categories searched on and the number of matched results for each category. If the number is greater than zero, then that number is displayed as a link to launch a new tabbed Web page with further details of that specific category. If the user selects this link, the GST then submits another request to the server to retrieve a general list of the initial set of matched results. Thus, the GST retrieves more extensive information only as the user requests it.

The newly created tabbed Web page includes a column of checkboxes for each match in the search. As a checkbox is selected, the JavaScript client issues yet another query request to the PHP server-side application, this time for detailed information on each matched result, and a table is created to the right of the Web page with a new row for each check box selection. The example shown in Figure 3 is for a search done on the string 'safety' in major category AOI. The matched search string results are highlighted with a yellow background and bold font.

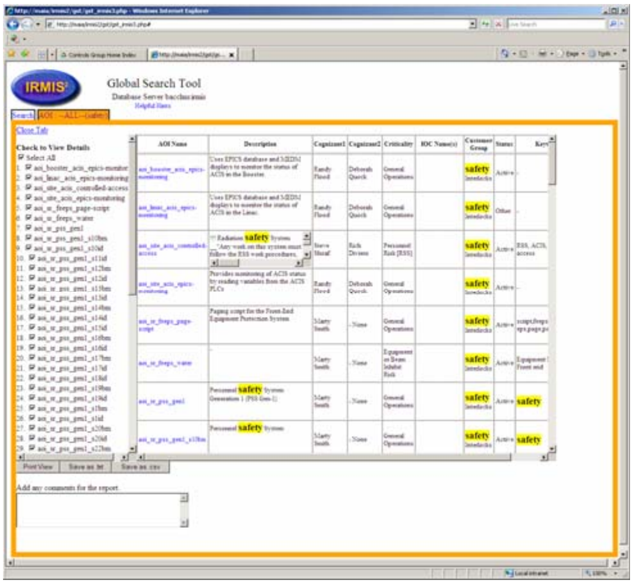

Figure 3: GST detailed search results.

Other features of the Global Search Tool include:

- EPICS process variables are not included in the default top level search of all major categories due to the large number of process variables in the IRMIS database (over 400,000) and the delayed search time required for this category. The Advanced Search button must be used to reach the option to search process variables.

- If there is a discrepancy between the initial data search and the general list data search for a given category, this is an indication of missing data in the IRMIS database for related attributes of a category. The GST writes information on this incomplete search result to a log file for later IRMIS data inspection.
- Links to existing IRMIS PHP viewers are provided in the GST table of search results details.
- A report tool is provided to capture the search results to a printer-friendly Web page, a comma separated value (CSV) file such as Microsoft Excel, or to a text file.
- Helpful hints for using the GST can be viewed interactively on mouse-over action of a text link.
- The GST was built in a generic software framework with JavaScript objects and PHP functions so that it can be easily extended as needed for new major and minor categories.

Performance

One of the main user requirements for the IRMIS Global Search Tool is a quick response time. Due to the large volume of data available in IRMIS, this presented interesting challenges in the use of AJAX technology. Rather than following the typical Model-View-Controller (MVC) philosophy of decoupling data (model) from the user interface (view), a different approach was adopted in the GST. To speed up the response time to the user upon lengthy retrievals and sorting of a large number of results, the decision was made to add HTML code for checkboxes on the server side where PHP sorting of data returned from the MySQL database is performed. The PHP creates a string of HTML checkbox code that is embedded along with the actual IRMIS database queried results in XML elements. The XML elements are then returned to the client side JavaScript application as a single text sting and are subsequently parsed as a JavaScript responseXML document type. The JavaScript client does not need to further sort the returned data as the code for HTML checkboxes has already been added on the server side. The time savings in performing a sort on the IRMIS database queried results only once (in PHP server side) was significant – from several minutes to almost instantaneous updates to the user Web page.

Currently, typical response times for searches in the GST are:

- Process Variables: highest return of two minutes for detailed information on 7,794 matches
- AOIs: approximately 30 seconds to return detailed information on 681 matches
- Installed Components: approximately four seconds to return detailed information on 109 matches

Development Obstacles and Design Shortcomings

AJAX technology comes with its own set of shortcomings as a Web application. JavaScript and CSS are inherently highly susceptible and dependent on the browser type (e.g., Firefox or Internet Explorer). A lot of time and frustration was spent on developing code that was compatible among both browsers. The asynchronous nature of AJAX makes it difficult to stop client-side requests to the server once they have been sent. Also, since AJAX-enabled Web pages are dynamically enabled, the browser “back button” can produce unexpected results. A design shortcoming of the Global Search Tool is that new data from the server are requested from the client (pulled) instead of pushed from the server as data in the MySQL database changes. This means that the user must explicitly request updates by either making a new search request through the Global Search Tool or by refreshing the Web page.

Future Enhancements

Although the outcome of the Global Search Tool was successful, there remains much room for improvement and added features. Future planned upgrades include limiting the number of process variables that can be viewed at a time to speed up response time, changing the log file display format to XML and displaying as a Web page using Extensible Stylesheet Language Transformations (XSLT), and adding more minor categories to the major category EPICS Record.

CONCLUSION

The use of AJAX technology for creating a comprehensive search tool for the IRMIS database proved to be extremely beneficial. The rapid response time for querying the IRMIS database and displaying results back to the user upon every user keystroke is impressive. This project served as a useful prototype for deploying future controls Web-based applications in AJAX.

REFERENCES

[1] D. Dohan, “Decision Support Facility for the APS Control System,” ICALEPCS 2001, 271 (2001); http://www.jacow.org.

[2] D. Dohan, N. Arnold, “Connection-Oriented Relational Database of the APS Control System Hardware,” PAC 2003, 2327 (2003); http://www.jacow.org.

[3] T. Negrino and D. Smith, *JavaScript and AJAX for the Web*, Sixth Edition, Peachpit Press, Berkeley, CA 2007.