1999

Interlibrary Loan Form Java Programming and Direct Request

Craighton T. Hippenhammer
Olivet Nazarene University, chhammer@olivet.edu

Bryan Wilhelm
MicroSoft Inc.

Recommended Citation

Hippenhammer, Craighton, and Bryan Wilhelm. "Interlibrary Loan Form Java Programming and Direct Request." Journal of Interlibrary Loan, Document Delivery & Information Supply 9.4 (1999): 5-13.

Interlibrary Loan Form Java Programming and Direct Request

by
Craighton Hippenhammer
and
Bryan Wilhelm

Abstract

Olivet Nazarene University's Interlibrary Loan PRISM Transfer (IPT) program development has progressed through a number of revisions, including 1) a C-programmed CGI script that includes the new OCLC review file patron fields; 2) an all new Java 32-bit Windows application; and 3) a platform-independent Java applet that can be run inside Web browsers. Both of the Java versions will work with OCLC's Direct Request ILL functions using the ISO 10161 protocol. This article explains the reasons for and the advantages of the further IPT/ISO development and the programming issues that had to be digested before a Java version could be completed. ONU continues to provide their IPT/ISO development as freeware to the Internet and library communities.

Authors

Craighton Hippenhammer is the Access Services and Information Technology Librarian at Benner Library and Resource Center, Olivet Nazarene University, Kankakee, Illinois. Bryan Wilhelm is a senior Computer Science student and Java programmer.

Introduction

Every library in the country should consider automating the process by which their patrons request materials through interlibrary loan. Using electronic forms for ILL requests has eliminated 80 to 90 percent of all typing in the Interlibrary Loan Department at Olivet Nazarene University and half of the paper handling. There's no doubt that similar efficiencies and convenience can be gained elsewhere. At the very least, forms that send ILL request data to the email account of the ILL department can cut down on and/or eliminate the retyping of handwritten forms.[i] But those libraries that use OCLC's ILL processing module can have their forms send the ILL request data to OCLC via one of two methods: OCLC's ILL PRISM Transfer (IPT) or the ISO 10161 protocol.

ONU's Information Technology Librarian reported on the development of ILL Web forms using C-programmed CGI scripts that ran on UNIX servers in an article that appeared in these pages over a year and a half ago.[ii] The state of technologies used in libraries has changed considerably since then. Libraries using Windows NT servers and other non-UNIX operating systems could not use our development nor could libraries without at least a modicum of UNIX and/or C-programming expertise modify our forms for their use. The further development of Java versions of ILL forms explained here in this article solves those limitations, making it possible for the average technical, non-programming librarian to add IPT- and Direct-Request-enabled ILL forms into their interlibrary loan workflows.

As a service to the library community, the authors of this article have made all of

IPT/

their electronic interlibrary loan forms, IPT and Direct Request development available on the Web at http://olivet.edu/Departments/Library/ILL/ill.html. Further questions about the files posted there may be sent to the authors.[iii]

ONU-ILL Version 2.2

The previous article reported that processing interlibrary loan requests to OCLC's review file had a number of shortcomings. First, with the state of the technology at the time there was no way to marry the official OCLC record for the item requested to the data in review file fields provided by the patron. Second, the review file was being used for more purposes than just receiving IPT data, which was causing confusion to some ILL department workflows. And lastly, the IPT process sometimes sent sensitive information, such as patron addresses, phone numbers and credit card numbers, to the review file that then got forwarded to the lender.

In the interim, OCLC improved the IPT process and addressed these shortcomings.[iv] They created an "apply" command that overlays the OCLC union record onto the review file request, matching the patron request to the official OCLC record and the item's OCLC number. This "ap" command makes processing requests from one's own patrons smoother and easier than before the command was created. Simply call up the review file record needed for processing, search the item, type the command ap, modify the custom-holdings-supplied lender string if needed, press <F10>, add a constant data name and produce the request.

Second, OCLC created an additional "save" command, which in effect created a new workfile for ILL staff to save work for workflow purposes. This enables the review file to be used solely for ILL requests from one's own patrons.

To solve the privacy issues, OCLC created twelve new patron fields in the review file, some of which can be viewed by the borrower but hidden to the lender. Credit card numbers, phone numbers, etc., can now be kept local.

Version 2.2 of Olivet Nazarene University's IPT program added most of these twelve patron fields and changed very little else. The CGI script is still programmed in C and still runs on UNIX computers. The program FTP's its ILL request data to an OCLC FTP account, which must be purchased from OCLC. The program does not work with OCLC's Direct Request.

The Java Versions

When Sun Microsystems released a new language called Java to the Internet in 1995, it was heralded as a new easy-to-use object-oriented programming language especially suitable to the Web and multiple platforms. OCLC has recently converted some of its utilities to Java, which enables OCLC users to adapt the online ILL process to the new language.

The two Java versions were created to increase the number of platforms on which the ONU_ILL program would run. Version 3.1, the Java application, runs on any platform that has a Java Virtual Machine. The compiled version (Ver. 3.11) that we have provided runs on Windows 32-bit operating systems. After our source code is tailored to match new situations and name systems, the resulting Java versions can also be compiled by Visual J++ Version 6 to run on Windows 95, 98, and NT as described in our online documentation.[v]

The Java interface varies from the previous versions in a number of ways. First, the user only has to fill out the patron information once and then may submit several requests without refilling out patron fields a second time. The program is set so each patron may submit ten requests before having to submit them to OCLC and having to start over with patron information again. This number, however, may be changed to whatever the library's maximum limit might be per session, including virtually unlimited numbers.

Second, the Java interface uses Windows dialog boxes for input, and required fields are emphasized by putting them up front. Optional fields appear in a secondary window by clicking on a button that reads, "More."

Lastly, the need-before date field is more intuitive and has some error-checking features that the first version did not. The field contains a default date that is twenty-five days from the current date that the patron may use or not.

Version 3.2 is a Java applet, which runs inside of a Web browser. It is not entirely operating system independent since it relies on the accessibility of a "redirect" utility

IPT/

running on the server as explained in more detail below. Included on ONU's IPT/ISO Development page are REDIR programs for Windows NT, Linux, and Solaris.

The Administration Module

In addition to the interlibrary loan form Java programs, we created an administrative module to make it easier to change customizable features of the application. The customizable features of the applet are controlled by the easily accessible parameters in the HTML file. The administrative module allows the changing of method of submission (data goes to the library's OCLC review file, to a preset lender string, to the first lender via a library's Direct Request profile, or to a simple email address), changing the Web addresses of the library's online catalog and the library's periodical holdings list, changing the default maximum number of requests, and changing the default need-before date.

For complete instruction on how to customize ONU ISO development to your library situation, follow the steps in the README file included at the ONU Web download site. The steps are laid out such that non-programmers should be able to make the necessary adjustments.

FirstSearch and Direct Request

OCLC's FirstSearch database interfaces include interlibrary loan functions that are connected to OCLC's ILL processing module (the message file's "review file"). In other words, the data the FirstSearch user types into the ILL Request Form is sent via a particular

IPT/

“sending” protocol to OCLC and ends up in the Review File. The original protocol that the FirstSearch ILL requests used when it was first created was a process called IPT: Interlibrary Loan PRISM Transfer, which included the FTP protocol, but OCLC recently moved to a new protocol called ISO 10161 which provides support for the control and management of ILL transactions for both borrowing and lending activities.[vi]

The ISO 10161 protocol is needed to make OCLC’s newest interlibrary loan service, Direct Request, work.[vii] Direct Request is a direct ILL process, from patron to lender, that bypasses borrower processing.[viii] The borrowing interlibrary loan department must have constant data, custom holdings and Direct Request profiles set up, and if lenders are found, if the maxcost fields of the borrower and the lender line up, then the request goes through directly to the first lender in the lender string. Issue information for periodicals cannot yet be retrieved automatically, so most of the Direct Request orders at this point are for books.

ONU Java programs also use the ISO 10161 protocol and therefore are Direct Request enabled. Since the ISO protocol is integrated into FirstSearch, the ONU Java forms would best be used for the interlibrary loan ordering of materials from citations retrieved in non-FirstSearch databases.

Version 2.1 CGI Script Programming Issues

OCLC’s recent upgrade to the Review Form and IPT process involved adding twelve new patron fields. Since we had already dealt with the difficult issues in programming an IPT module in C, adding a few more fields did not prove difficult. In Version 2.0, we had included most of these new fields, but had no unique fields to send them to. So we had to concatenate them (double them up) in other fields that already existed. The new fields actually simplified the programming because each html form field corresponded to the new IPT fields.

Version 3.1 Java Application Programming Issues

Versions 2.0 and 2.1 of the ONU IPT development used FTP to send the HTML field contents to OCLC. This worked with our C programming because UNIX can send the file name and its contents together. We hoped to write Version 3 in Java. Java, however, creates files on the remote server, but writes the contents to files afterward. The problem is that OCLC's server changes the name of the file (for indexing purposes and keeping file names unique), so Java doesn't know where to write the contents. Since Java could not send the file name and the contents together, we could not proceed with the IPT/FTP method.

The alternative was to use OCLC's new ISO utilities, but they were extremely large, complicated, and written in C (not the easiest language to understand). Unfortunately, OCLC had not yet supplied a Java version of these utilities to their users, so translating them from C to Java would be necessary if we were to proceed. After studying the ISO utilities for awhile, we made the conversion to Java, and once that barrier was hurdled, programming proceeded fairly smoothly.

Version 3.2 Java Applet Programming Issues

An applet is a Java program that runs within a browser. In addition to having the same problems as the Java application we mentioned above, applets have additional complexities that are difficult to overcome. Many of these restrictions are due to security concerns. For example, applets cannot make network connections to servers other than to their own host. An ILL form that submits its data to a server at OCLC would have to make a network connection to it. That cannot be done because of applet security. Our solution to this problem was to send our ILL data streams to our local server and then redirecting them to OCLC. Euphemistically, this is called a work-around.

This can be accomplished by a redirect command, REDIR, which we have made available on our site for a number of different platforms. The REDIR utility needs to reside on the host server, needs to be automatically activated, and needs to run at all times. It has three parameters: a listen port number (where we send the data on our host server), a target address (OCLC’s Web address) and target port (where the data arrives on OCLC’s server). The applet thinks the data is being sent to its own server, when in reality it is also being forwarded on to another server.

Problem number two: Applets cannot create files. The IPT method requires file creation, and had we continued trying to program along those lines without another creative “work-around,” the applet would not have been possible. Going with the ISO method kept us from having to examine this issue very closely since the ISO protocol does not require creating files.

In Versions 2.0 and 2.1 we used the UNIX command .netrc to create a unique user by retrieving the IP address, but applets cannot access IP addresses. So instead, we found another way: versions 3.1 and 3.2 use the library symbol followed by the current time in milliseconds, located in the transaction qualifier field, to uniquely identify each ILL request that is submitted.

The version of Java also proved to be a problem: the OCLC BER utilities had been translated from C into Java 1.1, but no browser at that time supported Java 1.1. After we had been working on the application for awhile, a Java 1.0 version of the BER utilities suddenly appeared during the summer months of 1998. Both are now available.

The ONU_ILL applet runs in both Netscape 3.0 and Internet Explorer 3.0. However, we ran into some interesting difficulties with Netscape. Netscape saves applets in the running state, meaning that data in applet fields will still be there after the back button is pushed (a privacy issue), but Explorer will restart the applet from the beginning if the back button is clicked (preferred). There were other irritating but non-lethal problems. The tab doesn’t work in applets within Netscape and graphics support is not consistent: e.g., our logo could not be drawn inside of dialog boxes.

Summary

Improvements to the IPT process have arrived, and all libraries using OCLC ILL processes now have an excellent electronic tool to process their ILL requests online. In fact, the technology now exists for a patron to request a book in WorldCat and have it arrive through interlibrary loan channels without the ILL department having to search for it and order it. The ILL department will have to handle the book when it comes in and see that it gets to the patron. FirstSearch-found books that do not meet the Direct Request criteria and all periodical articles at this point still have to be fully processed by the ILL department. But non-FirstSearch ILL requests will have to be entered into an electronic form that is connected to OCLC's Interlibrary Loan Transfer process via the ISO 10161 protocol for the Direct Request to work. Version 3 of ONU_ILL provides this extra step.

At this time, no database provider has been able to add the IPT/Direct Request process to their software other than OCLC itself.[ix] Hopefully, with the freeware ONU is providing, that difficulty can be hurdled as well as making it possible for colleges and universities world-wide to add IPT- or ISO-enabled online forms to process their interlibrary loan requests.

BIBLIOGRAPHY

1. Deitel, H. M., and P. J. Deitel. Java: How to Program. Second Edition. Upper Saddle

River, New Jersey: Prentice Hall, 1998.

2. Lemay, Laura. Teach Yourself Web Publishing with HTML 3.2 in a Week. 3rd edition. Indianapolis: Sams Publishing, 1996.

3. Newman, Alexander, et. al. Using Java. Special Edition. Indianapolis: Que Corporation, 1996.

NOTES

[i] Wang, Selina, and Glenn Manino, “Creating Interlibrary Loan Request Forms on the Web,” Journal of Interlibrary Loan, Document Delivery and Information Supply. 8.1 (1997): 79-86.

[ii] Hippenhammer, Craighton T. and Keith Reel, “Interlibrary Loan Web Forms and IPT,” Journal of Interlibrary Loan, Document Delivery and Information Supply. 7.3 (1997): 35-44.

[iii] Email Prof. Craighton Hippenhammer at chhammer at olivet.edu. This article and the documentation that is posted online is the extent of the technical help that the authors can provide.

[iv] OCLC. “Technical Bulletin 222 Revised: OCLC Interlibrary Loan Enhancements.” July, 1998 rev. http://www.oclc.org/oclc/tb/tb222/tb222.htm. September 7, 1998.

[v] Hippenhammer, Craighton T., “Interlibrary Loan: ILL PRISM Transfer.” http://www.olivet.edu/Departments/Library/ILL/ill.html. September 7, 1998.

[vi] Interlibrary Loan Application Standards Maintenance Agency. “An Introduction to the ISO InterLibrary Loan Application Standards.” 1997. http://www.nlc-bnc.ca/iso/ill/standard.htm. September, 8, 1998.

[vii] OCLC. “Guided Tour: ISO Connection.” http://www.oclc.org/oclc/drill/drisocon.htm. September 9, 1998.

[viii] OCLC, “OCLC ILL Direct Request Planning Guide.” May 14, 1998, rev. http://www.oclc.org/oclc/specs/dillspec.htm. September 7, 1998.

[ix] OCLC person contact.