LOCAl: A Flexible Web-Based Microscope Reservation System
Kevin W. Eliceiri, Vincent P. Chu, and John G. White
University of Wisconsin, Madison, WI, USA

Many microscopes, especially confocal and electron microscopes, are costly to purchase and maintain and are often only available in centralized facilities. The Internet has provided the potential for real-time accessibility to these microscopy resources with HTML pages describing the facility and the equipment available and, in some cases, the remote use of the microscope itself. Web-based booking systems have become another powerful way to improve accessibility to a microscope. A public-domain, calendar-based booking system for the online scheduling and user management of microscopes in laboratories and multi-user microscopy facilities is described.

One extremely powerful application of Internet technology has been for individual research laboratories to create an online presence through the Web. Establishing such a Web presence has been facilitated by the advent of “what you see is what you get” HTML editors and easy-to-use Web server applications. The vast majority of laboratories now have Web sites that provide information about the laboratory, such as research projects, favorite protocols, and access to publications, images, and movies. Reflecting a trend seen in the general commodity Internet, these scientific sites have now become more sophisticated, often containing interactive content such as forms and Web databases. These interactive elements are created with “Internet-aware” scripting languages such as Perl, PHP, and JavaScript. The use of these scripting languages is not new. Perl, for example, has been used for Web-based input forms for years. However, as these scripting languages have become more powerful, so has their potential.

About five years ago, we developed a Web calendar system for microscopes at the University of Wisconsin. Originally, we set out to provide a simple, no-frills, “paper calendar” system that would operate exactly like a paper calendar hanging on the door of a laboratory, except this would be modifiable and readable from a Web browser. As users and laboratory managers became accustomed to this simple system, we added new features such as user accounts, simple billing from the calendar, and the implementation of microscope usage rules in the calendar, such as hours of allowable use for different classes of user competencies.

As the system became more complex and the underlying code more unwieldy, we decided to write a new purpose-built scheduling application for microscopy, rather than continuing to modify simple text-based or “flat file”-based systems. We have used the Web scripting language PHP and the database system MySQL to create a booking calendar system for microscopes. The speed and flexibility of these Web scripting languages has led to them being used extensively over the last 10 years for the creation of reservation calendars. Commonly used for booking lecture halls or announcing such events as...
seminar series, these booking systems are being adapted for more specialized purposes such as the reservation and management of shared resources.

Our aim was to move toward a modern architecture in which a powerful database drives the system. Major requirements for the scheduling system were flexibility and for it to be easily customizable to a variety of applications. These goals were achieved in the following ways: (i) the complete program is not only freeware but also Open Source under the GPL agreement; (ii) the entire system is based on a software foundation of Internet standards (i.e., MySQL and PHP); (iii) except for the initial setup of the system, all configuration is done via an administrator’s Web console; and (iv) the system is database driven; all events and actions are initiated/accessed from within the database.

Out of this effort, we produced the LOCal calendar system (http://www.loci.wisc.edu/calendar), which has the appearance on-screen of a traditional monthly paper calendar. This is a two-tier system: a PHP front end on top of a MySQL database. This means little to the end user, except that the interface is fast, compatible with all major Web browsers and operating systems, and has the ability to sort, search, and modify all user data and events. However, from a systems administrator point of view, the use of PHP and MySQL was the result of a deliberate decision to provide a system that could grow and expand with the changing requirements of a facility. In addition, we wanted to use popular, standardized components and in no danger of being rendered obsolete in a few years.

Both PHP and MySQL have a strong development community and are widely used. PHP is a scripting language that allowed the creation and editing of dynamic Web pages in the same way regular HTML pages are created. Developed specifically for use with the Web, PHP has a greater degree of simplicity and elegance than Perl for this application. This allowed us to implement the system in succinct, maintainable code, while keeping the number of files as low as possible. In addition, compared with commercial Web solutions such as ASP and Coldfusion, PHP is just as robust and reliable, while being available at no cost or royalty fees.

MySQL was chosen to be the heart of the LOCal system, as it more than met our current needs and allowed for future expansion. MySQL is a very fast, multi-threaded, multi-user, and robust structured query language (SQL) database server. It supports most common language interfaces, including the C, C++, Java, Perl, PHP, Python, and Tcl Advanced Programmers Interfaces (APIs), which means that the API can be easily changed in the future. MySQL works on several different platforms and server applications, including Solaris® (Sun Microsystems, Palo Alto, CA, USA), MacOS®, Microsoft Windows NT/2000®, and Linux®.

Figure 2. Event modification window. This window allows the administrator to easily view all events for any given period and modify or delete them. The administrator can also reserve blocks of time for periodic events, such as a class that will use a microscope every Friday afternoon for a semester.
The LOCal system interface is simple with one entry URL for day-to-day users and another URL for calendar administrators. There are several significant features of LOCal: (i) a clean and simple interface with an integrated user database; (ii) fast loading and compatible with Netscape® and Explorer on any operating system; (iii) a detailed Description Window that not only has contact details but also has information on experimental details or instrument mode; (iv) advanced Administrator Functions including an instrument administrator alert for both new users and microscope signup; there is also an administrator Control Window where the administrator can remove any user, delete, or add any time slot and authorize new users for the system; (v) billing and invoicing system: users can be invoiced directly in the system; (vi) a privilege and password system that is very flexible and secure because all password traffic is encrypted; (vii) open-database-connectivity (ODBC) support that allows easy integration into existing laboratory databases such as Microsoft Access® and Filemaker Pro®. For example, an administrator could use Microsoft Access to connect to the MySQL server to synchronize with an existing user database with the calendar user database.

The major components of the system are a one-time initial setup, administrator functions, and user functions.

Initial setup of the system is performed once only by the server administrator. In general, this involves configuring four components: an Apache Web server for serving out the Web pages, PHP for interfacing to the database, the MySQL database, and the LOCal code. The setup of the Apache, PHP, and MySQL components is straightforward, as all three can be installed in a standard Linux installation or downloaded separately. In the future, we plan to support other server operating systems including MacOSX® and Win2000. Installation of LOCal is done from a tar file; eventually, we plan to make a LOCal RPM (redistribution package modules that simplify
installation on the Linux platform).

The administrator has four major functions at their disposal: (i) monitoring the calendars: the ability to modify, delete, and change calendar permissions for any given user (Figure 1); (ii) the modification of any event or the addition of any universal event such as recurring usage of a microscope (Figure 2); (iii) the addition of new calendar groups and changing the parameters for existing calendars (Figure 3); and (iv) invoicing and billing for any calendar users or any given calendar (Figure 4).

The user has three major functions: (i) the initial signup and creation of the account; (ii) requesting permission to use a calendar (this can be done upon creation of a new account or anytime from the user menu); and (iii) modification of user data, including the ability to change parameters such as contact info, password, etc.

We have recently completed an extensive testing and debugging phase, and the LOCal system is now successfully in use at two major microscopy laboratories. We plan on distributing LOCal via our Web site and will integrate it into the installer already available for other programs from our programming group. In addition to the features mentioned above, we welcome suggestions for enhancements and improvements that will be of interest to the microscopy community for possible consideration for implementation in the next release of LOCal.

Planned features for the next release include additional invoicing capabilities and installer versions for Microsoft Windows™. In addition, at the request of several laboratories, we are in the process of creating an interface application that could run on a local computer attached to a microscope and could be used to synchronize with the LOCal database. This would allow tracking of actual usage of a microscope rather than signed-up usage and, if desired, could also be used to restrict access to a microscope if the user was not signed up on the calendar.

More information on the LOCal system is available at our Web site at http://www.loci.wisc.edu/calendar.

Address correspondence to Dr. Kevin Eliceiri, LOCI, 1675 Observatory Drive, University of Wisconsin-Madison, Madison, WI 53706. e-mail: eliceiri@facstaff.wisc.edu

Suggestions for contributions to the BioImaging feature are welcomed by its editor, Dr. Steve Paddock (paddock@facstaff.wisc.edu)