TECHSIGHTING
SOFTWARE

Just a Circle Won't Do

When a plasmid construct arrives in the mail, the recipient usually wants its restriction map. The Internet provides ready access to DNA sequences, but most programs do not provide a convenient interface to this information. A drawing program can be used to draw a map, but this approach requires the user to scale the maps (in base pairs per inch) and manually identify restriction sites. The result is often not much better than a hasty scribble on a note pad. Programs are available that create attractive plasmid maps; however, most of these are high-end applications that are expensive, cumbersome, difficult to learn, and poorly integrated with the Web.

Plasmid 1.0
Redasoft, Inc.
Toronto, Ontario
Canada.
$199.
416-656-9617
www.redasoft.com

Recently, a new software product called Plasmid 1.0 was released. It opens DNA sequence files and automatically converts them into plasmid maps. It will also allow one to draw a map independent of sequences. Two versions of the program are available, one with a software interface to the Internet and one without. Within minutes, a user with basic computer skills can generate detailed maps of plasmid constructs. Upon selection of the "NEW" command from the menu or toolbar, a dialog box appears, asking how many base pairs make up the total sequence. A scaled "circle" or bar is subsequently presented, ready for detailing. It is easy to insert regions of interest on the plasmid map by using the pull-down menus of the tool bar and entering the base pairs to be spanned. The REBASE tool identifies restriction sites in map sequences automatically; alternatively, users can define the location of restriction site markers manually on a map in the absence of sequence data. All boxes, lines, and names can be customized for font, font size, color, style, and line thickness. Users may also enter text labels. The resulting display is ready for saving, printing, or importing into documents created by other programs. No manual accompanies Plasmid 1.0. All help is provided online.

Perhaps, the most advanced and interesting feature of Plasmid 1.0 is the ability of the Web-browser version of the program to read sequences from the Internet. Plasmid 1.0 is designed for use with Netscape Communicator or Microsoft's Internet Explorer. Users can access Internet databases containing DNA sequence information by selecting "Browse the Web" from the Tools menu. Popular sequence database addresses (for example, National Center for Biotechnology Information database at the National Institutes of Health) are preset in the program, and other URLs may be added. Selecting "Browse the Web" divides the Plasmid 1.0 window in two, with the right side displaying the Web site and the left side containing the map. Users can easily copy and paste information for making the map as necessary. Other options include simple map drawing independent of the clipboard, via the "New from Web" menu option.

Plasmid 1.0 can be purchased and downloaded from the Redasoft home page (www.redasoft.com). The company is Web-based, with all ordering and support conducted over the Internet. Their site is very informative and provides product information, known program conflicts, and a free demo version of the program. Work is in progress to repair minor bugs and update the commercial product to include more features, such as view zooming and restriction enzyme site lists. At a minimum, Plasmid 1.0 requires a small amount of hard disk space and 8 MB of RAM on PC-compatible machines operating Windows. No Macintosh version is available.

Plasmid 1.0 is a simple, relatively inexpensive utility for making plasmid maps. The program's strengths lie primarily in its integration with the Web. Plasmid's other features provide only weak competition for other, dedicated plasmid mapping products.

—FRANK HOOVER

Department of Physiology, Institute of Basic Medical Sciences, University of Oslo, Post Office Box 1103, Blindern, 0317 Oslo, Norway. E-mail: frank.hoover@basalmed.uio.no

TECHSIGHTING
SOFTWARE

Digital Duct Tape

As labs increasingly embrace automation and computer technologies, the need for tools to manage the growing amount of data becomes more and more important. Usually, data management requires a database to serve as a repository that can be searched, saved, and edited. However, getting data into a database can be a challenge in a laboratory setting, where information can come from many disparate sources. What is needed is an all-purpose tool to move data and manipulate the computers that store the data.

Enter Perl, a powerful programming language that has become known as the "duct tape" of the Internet. Perl's ability to glue together data from just about anywhere is useful to a wide array of applications, but there are five features of Perl that make it particularly attractive for use in a biomedical laboratory. First, Perl code will run on the three major operating systems commonly found in the lab: Windows (NT and 95/98), Macintosh, and Unix (Linux). In fact, the exact same Perl code can often be simply moved from one operating system to another and run without change. Second, the basics of Perl are relatively easy to learn. You write Perl in a simple text editor, and no special tools are required. When you run a Perl program, you just feed the code to a software engine that interprets your commands and does whatever you have coded. Third, Perl is one of the best tools available to manipulate text. In fact, it has an entire internal language for pattern matching and text substitutions. Fourth, hundreds of Perl modules (or prebuilt pieces of code) exist on the Web. You can find code that makes graphics, manipulates text, runs administrative routines, and so on. And finally, Perl is supported on the Web by an open community of developers who are constantly providing new features and answering questions from experts and novices alike.

Perl can help in the lab in various ways. Suppose you have reams of digital data from a scintillation counter run and you want to move all of this into an Access database for storage or transfer to a colleague. You could cut and paste all of the data into the proper fields, but that would be labor intensive and error prone. Instead, you could write a short Perl program that opened up each file, scanned for header elements and data, and entered the data into individual records in the database. The exquisite pattern-matching abilities of Perl make parsing text a straightforward task. Another example of pattern matching might be to sort through your old e-mail messages. If you use Microsoft Outlook Express as a client, your mail is stored in a special text file in the Windows/Application Data directory. With Perl, you could read the text of your mail, parse out the From:, To:, and Body: fields, and enter them in a database for safekeeping.

Another example of Perl in the lab might be to manipulate DNA sequence data. In fact, if you look behind the scenes of a major sequencing lab, you are likely to find Perl tools. With Perl, you can scan rapidly through thousands of lines of sequence to find absolute matches and pattern matches—from simple, such as Type II endonucleases, to complex, such as degenerate repeat motifs.

Perl can also talk to the operating system itself. If you use Windows NT, for ex-

ample, you could completely automate the entry of passwords and file permissions, using Perl "administrative" modules that are easy to program. Just ask anyone running even the smallest of LANs (local area networks) how tedious system administration issues can be when you run a server, and you will see the value of a Perl solution.

Perl can also be used to generate Web pages to publish data on a lab Web site. Suppose you have data on yeast strains and genotypes that are stored in an Excel spreadsheet. You would like to make Web pages to show your colleagues what strains you have and how to request them. You could use Perl's ability to talk to Windows applications to read the data from the spreadsheet and generate HTML code for the Web on the fly. In fact, because Perl can also FTP (file transfer protocol) documents across the Web, you could automatically FTP the Web pages to your Web server.

And finally, because Perl works well with Web servers, you can use simple Perl scripts to capture all kinds of data through Web forms. You could use Web forms to get registrations for a meeting, requests for mouse strains, or even have potential postdocs upload resumes for your review.

We have tried to touch on a few examples of Perl applications in a lab, but ultimately the best applications will likely come from interested graduate students or postdocs who see a problem and have a basic understanding of the Perl language and tools. The Internet can serve as their training ground. On the Web, you can find samples of code, software, and tips from a world of experts. We suggest starting at www.perl.org and www.activestate.com. The first URL is for anyone interested in the basics of Perl, and the second is a great one-stop shop for Windows users only. Virtually all of the tools you need to use Perl are absolutely free on the Web.

—ROBERT SIKORSKI AND RICHARD PETERS

TECHSIGHTING
BIOCHEMISTRY

Dracula's Wish

Heparin, which was discovered in 1916 (*1*), binds and activates antithrombin III. The complex of heparin–antithrombin III inhibits a number of blood coagulating factors (thrombin and factors Xa, XIIa, XIa, and IXa), but the mechanism of inhibition is somewhat different for thrombin than for the other coagulation factors. For the other factors (most importantly, factor Xa), the binding of heparin to antithrombin III leads to a conformational change in the inhibitory loop of antithrombin, which results in the recognition and inhibition of the active serine centers of the coagulation proteins. In the case of thrombin, the heparin-antithrombin complex electrostatically binds and inhibits the protein.

This difference in mechanism of inhibition between thrombin and the other coagulation factors explains why smaller heparin molecules can still inhibit factor Xa, but fail to inhibit thrombin. Preservation of thrombin inhibition requires a longer chain attached to heparin in order to provide the necessary electrostatic attraction. It is this very electrostatic charge, though, that is often responsible for the side effects of heparin. Indeed, this large polysaccharide (mean molecular weight of 15,000) has a tendency to be very "sticky" and binds a number of other biological molecules, including platelet factor 4 (PF4). Antibodies recognize PF4-heparin complexes and activate platelets, leading to thrombocytopenia as well as arterial and venous thrombi. Anyone who has spent a little time in clinical wards knows that heparin-induced thrombocytopenia is a significant cause of morbidity (and even mortality) in anticoagulation therapy. Advances in oligosaccharide chemistry has permitted the synthesis of the pentasaccharide antithrombin-binding site of heparin, which retains inhibition of factor Xa but lacks inhibition of thrombin. These truncated versions of heparin, also referred to as low molecular weight heparin (molecular weight of 4000 to 6500), have made their way into the clinical setting thanks to a decrease in this side-effect, because their lack of thrombin inhibition also results in a lack of activity against PF4.

The authors of a recent *Nature* article wondered whether they could synthesize heparin mimetics that would inhibit thrombin, but lack side effects (*2*). This is not a trivial feat, because the thrombin-binding activity is directly correlated with electrostatic charge, which itself leads to undesired interactions with other proteins. A careful analysis of heparin indicates that a synthetic compound that could mimic heparin's full anticoagulant activity would need an antithrombin-binding domain (A domain) and a thrombin-binding domain (T domain). The authors used saccharides where the *N*-sulfate groups are replaced by *O*-sulfate groups and hydroxyls are alkylated; these modifications of the saccharide unit greatly simplify the synthesis process.

The authors first tried to synthesize polysaccharides consisting of repeating units of the A domain. Because the A domain is also negatively charged, they postulated that a sufficiently large polysaccharide might bind thrombin as well as antithrombin. Indeed, saccharides of 16 units and larger were able to inhibit thrombin both in vitro and in an animal thrombosis model (arterio-venous shunt in the rat). Smaller saccharides lacked thrombin inhibition, but still retained anti-factor Xa activity, because the A domain can bind to antithrombin. All the molecules, however, cross-reacted with PF4, and hence, had unacceptable side effects.

The authors then had no other choice but to consider linking an A domain to a T domain. But how should the units be oriented? Crystallographic data of heparin bound to antithrombin helped guide them to attach the T domain at the nonreducing end. For the A domain, a high-affinity analog of the antithrombin-binding sequence was selected (*3*). For the T domain, alternating α- and β-linked 3-*O*-methyl-2,6-di-*O*-sulfo-D-glucose units were used. The minimum number of saccharide units required to inhibit thrombin was 15. A 19-unit saccharide polymer was as potent as a standard heparin preparation. Once again, however, the synthesized compounds interacted with PF4.

The researchers then decided to try to decrease the electrostatic charge of the synthetic molecule by intercalating a neutral methylated hexasaccharide between the T and A domains. The resulting compound inhibited thrombin without affecting PF4. In addition, it was 5- to 10-fold more potent than standard heparin in in vivo models. This is probably due to the fact that this compound interacts much less with basic proteins, thus decreasing nonspecific binding and leaving more molecules available for binding to thrombin. More importantly, this compound seems to have little effect on platelets: it does not activate platelets in the presence of plasma from heparin-induced thrombocytopenia patients and it does not affect the bleeding time of rats while still inhibiting thrombosis.

This elegant work demonstrates how the use of synthetic chemistry guided by structural biology data and quite a few educated guesses can lead persistent scientists to the discovery of what appears to be a potent substitute for an important medical molecule, with none of its side effects.

—RICHARD PETERS AND ROBERT SIKORSKI

References

1. J. McLean, *Am. J. Physiol.* **41**, 250 (1916).
2. M. Petitou *et al.*, *Nature* **398**, 417 (1999).
3. P. Westerduin *et al.*, *Bioorg. Med. Chem.* **2**, 1267 (1994).

Tech.Sight is published in the third issue of each month. Contributing editors: Robert Sikorski and Richard Peters, Mednav.com, Byfield, MA, and Kevin Ahern, Department of Biochemistry and Biophysics, Oregon State University, Corvallis, OR. Send your comments by e-mail to techsight@aaas.org.