BioSubroutine: an Open Web Server for Bioinformatics Algorithms and Subroutines

Joowon Lee[1], Hana Kim[1], Wonhye Lee[1], Dongil Chung[1] and Jong Bhak[1,2]*

[1]BioSystems Dept., KAIST, Daejeon 305-701, Korea, [2]National Genome Information Center, KRIBB, Daejeon 305-333, Korea

Abstract

We present BioSubroutine, an open depository server that automatically categorizes various subroutines frequently used in bioinformatics research. We processed a large bioinformatics subroutine library called Bio.pl that was the first Bioperl subroutine library built in 1995. Over 1000 subroutines were processed automatically and an HTML interface has been created. BioSubroutine can accept new subroutines and algorithms from any such subroutine library, as well as provide interactive user forms. The subroutines are stored in an SQL database for quick searching and accessing. BioSubroutine is an open access project under the BioLicense license scheme.

Keywords: Bioinformatics subroutines, Bioperl, SQL database.
URL: http://biosubroutine.org and http://cellome.kaist.ac.kr/BioSubroutine/

Introduction

The first usage of the term 'Bioperl' was in 1994-1995 (Stajich *et al.*, 2002). It is an open-source and open-access project for re-useable programming (Lynex and Layzell, 1998) resources. Perl (http://perl.com) was an easy and convenient programming language for early bioinformatics projects including gene annotation and genome annotations. A library file named Bio.pl was created in which approximately 1000 bioinformatics subroutines were stored. The goal was to create a library of subroutines that are programmed in such a way that they can be easily accessed by other bio-programmers. Since then, an object oriented version of Bio.pl became an international resource of open and free biological programming. Bioperl (http://bioperl.org) is a powerful resource, and it has numerous subroutines and modules (Mangalam *et al.*, 2002). Alongside Bioperl, the original Bio.pl has been used as a subroutine library by a small number of programmers (http://bioperl.net). The idea behind the Bio.pl file is to provide subroutines in the easiest way so that programmers can borrow and modify them without using a larger object oriented module.

There have been other subroutine packages that were created for biological research, such as SPACAR (Van Soest *et al.*, 1992), a software subroutine package for simulation of the behavior of biomechanical systems and CCP4 suite (Collaborative Computational Project, 1994), a collection of programs for protein crystallography. The Biosubroutine package is for general bioinformatics research and unlike these other two packages it is available online, allowing easy access. Also, the package allows the modification and addition of subroutines by other users so anyone can easily contribute to the database.

As an extension of the original Bio.pl, we have constructed a relational database server for any type of bioinformatics programs that are not restricted to Perl. The main contribution is that we categorized subroutines that are used in bioinformatics research and placed them on an online server for easier access and usage. This is to avoid reprogramming the same or similar subroutines. Additionally, the server facilitates addition of subroutines, modifications, and maintenance through an HTML interface. Also, automatic parsing can be made difficult as a result of a number of different styles of adding comments in subroutines. In order to alleviate this problem, we created guidelines for updating and including comments about the subroutines. Although these guidelines may be bothersome, it is recommended that users follow them for the ease of other users. Finally, anyone who wishes to contribute to the database may do so easily through the HTML interface. Any changes made to the database will have to be approved by the administrator. While other subroutine packages were not easily modified, the Biosubroutine package has turned the Bio.pl file into a online open source project

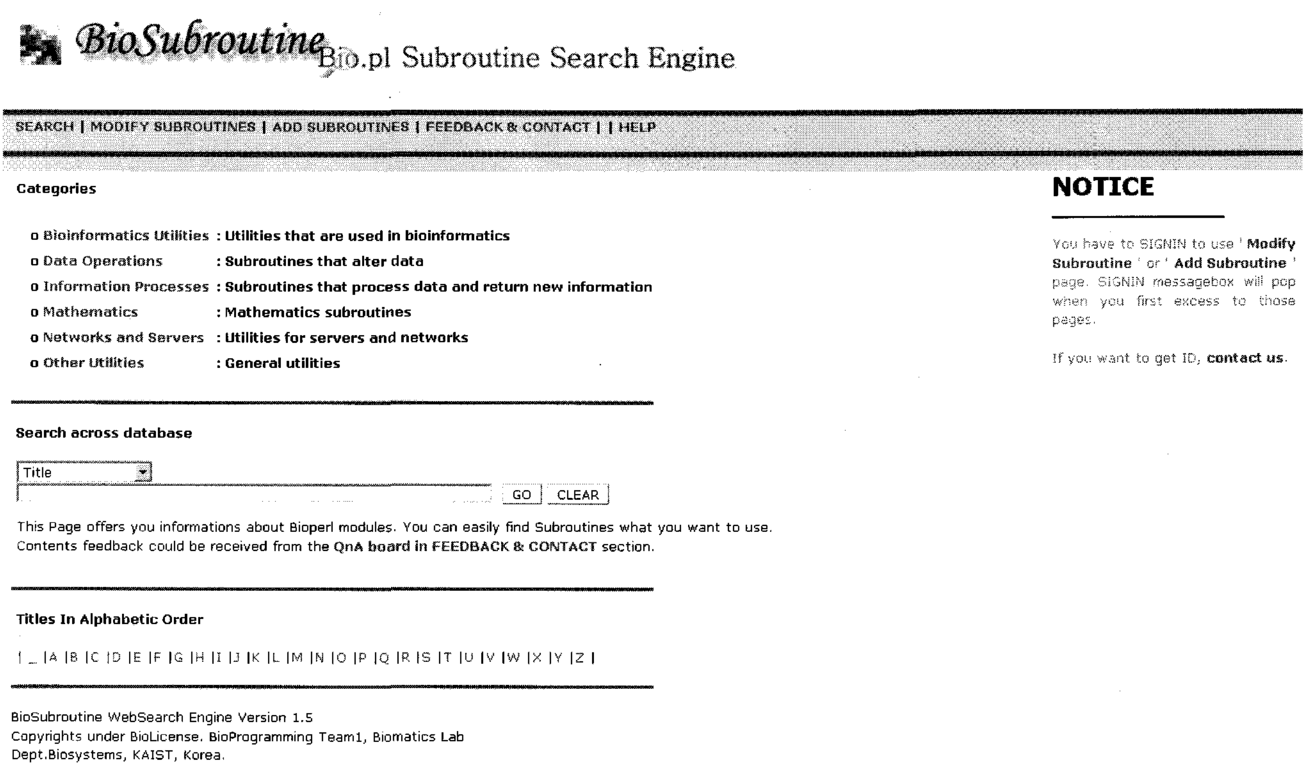
Methods

Database & Server construction

The original Bio.pl file was compromised of multiple

*Corresponding author: E-mail biopark@kaist.ac.kr, Tel +82-42-869-4318, Fax +82-42-869-4310

Accepted 15 February 2005


BioSubroutine
Bio.pl Subroutine Search Engine
SEARCH | MODIFY SUBROUTINES | ADD SUBROUTINES | FEEDBACK & CONTACT | | HELP
Title
GO
CLEAR

Categories

o Bioinformatics Utilities	: Utilities that are used in bioinformatics
o Data Operations	: Subroutines that alter data
o Information Processes	: Subroutines that process data and return new information
o Mathematics	: Mathematics subroutines
o Networks and Servers	: Utilities for servers and networks
o Other Utilities	: General utilities

Search across database

This Page offers you informations about Bioperl modules. You can easily find Subroutines what you want to use.
Contents feedback could be received from the QnA board in FEEDBACK & CONTACT section.

Titles In Alphabetic Order

| _ |A |B |C |D |E |F |G |H |I |J |K |L |M |N |O |P |Q |R |S |T |U |V |W |X |Y |Z |

BioSubroutine WebSearch Engine Version 1.5
Copyrights under BioLicense. BioProgramming Team1, Biomatics Lab
Dept.Biosystems, KAIST, Korea.

NOTICE

You have to SIGNIN to use ' **Modify Subroutine** ' or ' **Add Subroutine** ' page. SIGNIN messagebox will pop when you first excess to those pages.

If you want to get ID, **contact us**.

Fig. 1. BioSubroutine user interface.

subroutines with comment boxes before each subroutine. We parsed each comment field separately along with the source code and inserted the results into a MySQL database. Each comment field has its own MySQL field with the source code also compromising a single field. Adding, modifying, and searching of subroutines is done through HTML and Perl scripts that use MySQL internal commands.

Description of the database server and interface

The HTML interface for the server provides options for searching subroutines (Fig. 1), direct links to each of the subroutine categories, and adding and updating subroutines. Addition and updates of subroutines will have to be approved by an administrator before inclusion into the database in order to prevent inclusion of faulty subroutines.

There are four web pages: (1) the main and search page, (2) a subroutine addition page, (3) a subroutine modification page and a feedback link, and (4) contact page. On the main page there are the search functions and links to subroutines sorted alphabetically and by category. The subroutine addition and modification pages allow users to input or modify subroutine code or comments.

Subroutine searching uses MySQL internal search routines. When it receives an 'input string' from users, BioSubroutine gives the result by using that string as the query in MySQL. If the string is composed of multiple words, the spliced words are used for searching. The user can choose several options such as limiting the search area for only a particular field, not for all the information about the subroutine.

In the 'ADD_subroutines' page, the user can add new subroutines by following the form. For the improvement and clearness of our database, we strongly recommend that the users fill in the comments. By clicking the submit button, the user can check the contents that were added and can decide to modify the contents. The addition process is completed by using the 'INSERT' command in MySQL. To prevent the inclusion of faulty subroutines, administrator authentication will be needed for final inclusion into the database.

By typing the ID number of a certain subroutine on the 'Search' page, an authorized user can modify the subroutine contents. The subroutine contents are shown when the user types the ID, and clicks the 'submit' button next to the ID number text box. The user can edit any field s/he wishes to modify by clicking the 'EDIT' button right next to each field; Title, Function, Example, etc.

When the button is clicked, a new window with a text box appears, and the user can change the comments or the source code. To apply the modification into the original database, the 'UPDATE' command is used.

In order to insert the subroutines into the data table, we parsed the Bio.pl file comment tables and inserted each of the comment fields into each of the table fields using Perl regular expressions. There are 15 fields that correspond to the comment fields with two more fields for ID and source code. Every time a new subroutine is added, the ID is automatically increased so that addition is faster. There is a main table which has all the subroutines, and the ID is assigned in the order in which they were inserted. In addition to the main table, there are also subsidiary tables which contain the subroutines that have been categorized into the five aforementioned categories. There are also subroutines that can be looked up by alphabetic order, but these are done by searching the main table.

Categorization of subroutines

The categorization scheme for subroutines is based on the type of data used and returned. The descriptions of each category are as follows:

1. Utilities
   These subroutines do not deal directly with data that are used in bioinformatics and are used for general purposes.
2. Data Processing
   These subroutines utilize bioinformatics data and returns information about the data.
3. Data Operations
   Unlike Data Processing subroutines, Data Operations subroutines alter and return the altered bioinformatics data.
4. Mathematics
   This category is comprised of subroutines that perform mathematics functions.
5. Network/Server
   Subroutines that perform networks and server related utility functions.

Guidelines for the comment fields

The original comment field format, when properly submitted, can be very informative about the subroutines. However, irregular and blank comment fields can make it hard to use the subroutines. So we have developed some guidelines to follow when new subroutines are added to the database.

1. Users are asked to fill all the fields.
2. When naming new subroutines, users are asked to try to make the name into a short description of the subroutine's function. Words are to be separated by an '_'.
3. In this Version, field users are asked to put 'alpha' or 'beta' if the subroutine is buggy, and proper version numbers if it is finished.

Summary of the subroutines that compromise the database

There are over 1000 subroutines in the database, and a majority of them are designed for bioinformatics research. They were created by a number of people. The most important feature of the subroutines is that they share common comment field formats and name structure. Other features include that the subroutines, with the exception of a few cases, rarely call each other so that they are easier to use in other programs, and that the majority of subroutine names indicate their function.

Usage

Below the title drawing there is a bar with links to the main sections of the site: Search, Modify Subroutines, Add Subroutines, Feedback & Contact, and Help. The Search section, or the main page of the site, allows users to search for subroutines according to the following criteria: title, ID, function, usage, and keywords. Searching according to multiple criteria is also allowed. In addition to searching functions, the Search section also provides an index of subroutines according to classification and alphabetical order. The Modify Subroutines link sends the user to a page where subroutines can be called up and have their comments and source code modified. Modified subroutines have to wait for administrator permission to be included into the database. In the Add Subroutines section there are three methods of inputting comments and source code: input all fields in separate boxes or into a single box, or the option of uploading a file. There are limits on the size of uploaded files. The Feedback & Contact section has a contact list and a web board for questions. Finally, the Help section has a manual and tutorial for first-time users.

Results and Discussion

There have been reports regarding problems about using components from online repositories. Some problems are that people do not trust the quality of repositories, have an aversion to using other people's code, or find a lack of information about the components.

BioSubroutine is an open software component repository that enables addition and modification of Perl subroutines frequently used in bioinformatics research. It uses the BioLicense scheme for maximum exchange of information (biolicense.org). While Bioperl may provide more comprehensive resources, the BioSubroutine repository is easier to use and also provides subroutines that can be used in other fields other than bioinformatics. Also the policy of having an administrator overseeing addition and modification ensures problems, such as the lack of information regarding source codes and bug-ridden code, that frequently occur in unmanaged code repositories do not happen.

Acknowledgements

We are indebted to Moonsoul Jung for providing the funds for the BioSystems Department in KAIST. We thank BioSystems professors for their collaboration and teaching. This project was partly supported by R01-2004-000-10172-0 grant of KOSEF and Biogreen21 program of RDA. We thank Maryana Bhak and Daeui Park for editing this manuscript.

References

Collaborative Computational Project. Number 4 (1994). The CCP4 suite: programs for protein crystallography. *Acta Crystallogr D. Biol. Crystallogr.* 50, 760-763.

Lynex, A. and Layzell, P.J. (1998). Organizational considerations for software reuse. *Annals of Software Engineering* 5, 105-124.

Mangalam, H. (2002). The Bio* toolkits-a brief overview. *Brief Bioinform.* 3, 296-302.

Stajich, J.E., Block, D., Boulez, K., Brenner, S.E., Chervitz, S.A., Dagdigian, C., Fuellen, G., Gilbert, J.G., Korf, I., Lapp, H., Lehvaslaiho, H., Matsalla, C., Mungall, C.J., Osborne, B.I., Pocock, M.R., Schattner, P., Senger, M., Stein, L.D., Stupka, E., Wilkinson, M.D., and Birney, E. (2002). The Bioperl toolkit: Perl modules for the life sciences. *Genome Res.* 12, 1611-1618.

Van Soest, A.J., Schwab, A.L., Bobbert, M.F., and van Ingen Schenau, G.J. (1992). SPACAR: a software subroutine package for simulation of the behavior of biomechanical systems. *J. Biomech.* 25, 1219-1226.