Acta Crystallographica Section D
Biological Crystallography
ISSN 0907-4449

LISA: an intranet-based flexible database for protein crystallography project management

Peter W. Haebel, Vickery L. Arcus, Edward N. Baker and Peter Metcalf*

School of Biological Sciences, University of Auckland, Private Bag 92019, Auckland, New Zealand

Correspondence e-mail:
peter.metcalf@auckland.ac.nz

Received 9 April 2001
Accepted 5 June 2001

The increase in the number of projects carried out in protein crystallography laboratories has emphasized the need for effective management of project information and data. To meet this need, a flexible web-accessible database for protein crystallography project management (LISA) has been developed using the open-source software *MySQL* and *PHP*4. The database contains information about all aspects of structure-determination projects, including primer and plasmid sequences, protein expression, purification and crystallization results, structure coordinate files and resultant publications. The database web pages include links to relevant servers and contain, in addition, tools for processing stored information. The software package is freely available.

1. Introduction

It has been estimated that the number of known macromolecular structures could triple by 2005 (Berman *et al.*, 2000). Within 5 y, it is expected that the recently initiated structural genomics projects alone will contribute more than 1500 new protein structures per year (Thornton, 2001). The accelerated growth in macromolecular crystallography is a consequence of recent progress in structural genomics, incorporating high-throughput methods, and structural research projects growing in complexity. These latest developments are accompanied by a dramatic increase in the amount of data generated in protein crystallography laboratories.

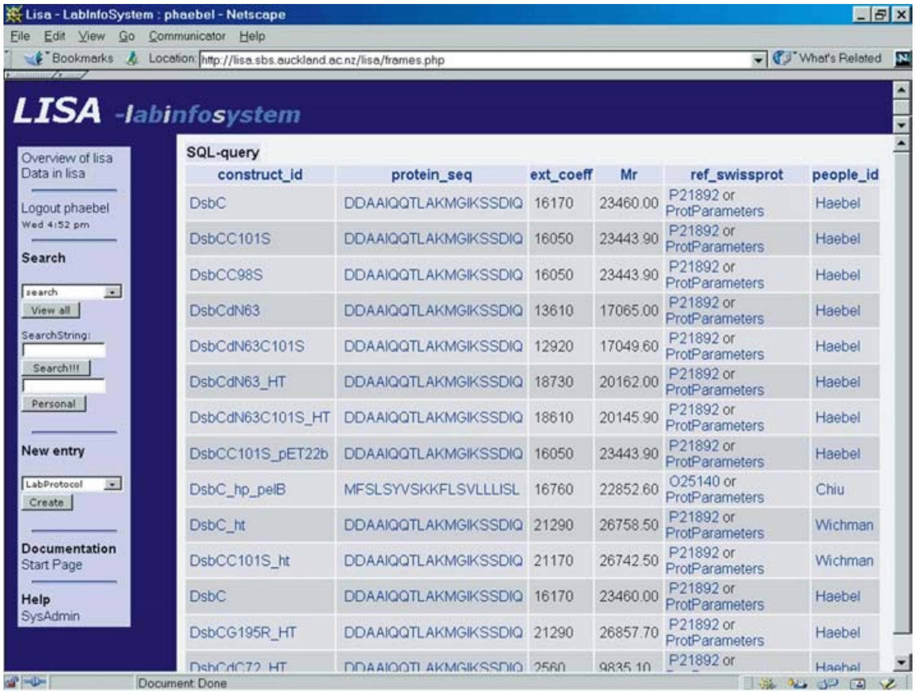
Many macromolecular crystallography groups face the problem of organizing the diverse range of data that make up a typical structural biology project. Information ranging from the results of preliminary database searches and bioinformatics procedures through details of PCR primers and expression plasmids, records of protein purification and crystallization experiments to the atomic coordinates of solved structures and associated diffraction data needs to be recorded and stored. In most laboratories, this information is recorded in laboratory notebooks or is stored in digital form on computer systems that control laboratory instruments. The diversity of information-storage media makes it difficult to retrieve and search laboratory information.

Laboratory information systems are designed to overcome this problem by storing information in a central relational database, where related items of information are logically linked. These information systems also include tools for searching and analysing data, useful for efficient data retrieval, summarizing projects and improving research strategy.

After investigating available laboratory information systems (Goodman *et al.*, 1998; Jakobovits *et al.*, 2000), we decided to develop our own web-based laboratory information system. The system, called LISA, is constructed from freely available software components including a web server and a widely used relational database server. Information stored in LISA can be readily accessed from any networked computer system *via* web browsers. Data is displayed together with links to related entries, relevant web servers and application software. The configuration of the database is very flexible, allowing users to adjust the system to their needs. In this laboratory, we use LISA in a multi-user multi-platform environment to manage protein crystallography projects and also for general purposes such as storing experimental protocols, maintaining the laboratory telephone book and for stocktaking laboratory supplies.

2. Implementation

LISA is an intranet/internet-based relational database that allows users to store, process, analyse and retrieve laboratory data using standard web browsers. Data can be entered and accessed from any networked computer, independent of its operating system. Access to the database is restricted to registered users through a user/password combination. Session management and data encryption by means of



SQL-query

construct_id	protein_seq	ext_coeff	Mr	ref_swissprot	people_id
DsbC	DDAAIQQTLAKMGIKSSDIQ	16170	23460.00	P21892 or ProtParameters	Haebel
DsbCC101S	DDAAIQQTLAKMGIKSSDIQ	16050	23443.90	P21892 or ProtParameters	Haebel
DsbCC98S	DDAAIQQTLAKMGIKSSDIQ	16050	23443.90	P21892 or ProtParameters	Haebel
DsbCdN63	DDAAIQQTLAKMGIKSSDIQ	13610	17065.00	P21892 or ProtParameters	Haebel
DsbCdN63C101S	DDAAIQQTLAKMGIKSSDIQ	12920	17049.60	P21892 or ProtParameters	Haebel
DsbCdN63_HT	DDAAIQQTLAKMGIKSSDIQ	18730	20162.00	P21892 or ProtParameters	Haebel
DsbCdN63C101S_HT	DDAAIQQTLAKMGIKSSDIQ	18610	20145.90	P21892 or ProtParameters	Haebel
DsbCC101S_pET22b	DDAAIQQTLAKMGIKSSDIQ	16050	23443.90	P21892 or ProtParameters	Haebel
DsbC_hp_pelB	MFSLSYVSKKFLSVLLLISL	16760	22852.60	O25140 or ProtParameters	Chiu
DsbC_ht	DDAAIQQTLAKMGIKSSDIQ	21290	26758.50	P21892 or ProtParameters	Wichman
DsbCC101S_ht	DDAAIQQTLAKMGIKSSDIQ	21170	26742.50	P21892 or ProtParameters	Wichman
DsbC	DDAAIQQTLAKMGIKSSDIQ	16170	23460.00	P21892 or ProtParameters	Haebel
DsbCG195R_HT	DDAAIQQTLAKMGIKSSDIQ	21290	26857.70	P21892 or ProtParameters	Haebel
DsbCdC72_HT	DDAAIQQTLAKMGIKSSDIQ	2560	9835.10	P21892 or	Haebel

Figure 1
The graphical user interface of LISA. The interface has two panels to facilitate navigation and display data. The navigation panel (left, blue background) provides access to most functions of LISA. Retrieved data is displayed in the results panel on the right. Each row in the results panel represents one database record, *e.g.* information about a protein, including individual attributes such as protein sequence and molecular weight. Hyperlinks (blue text) lead to related information in the database or to relevant web servers.

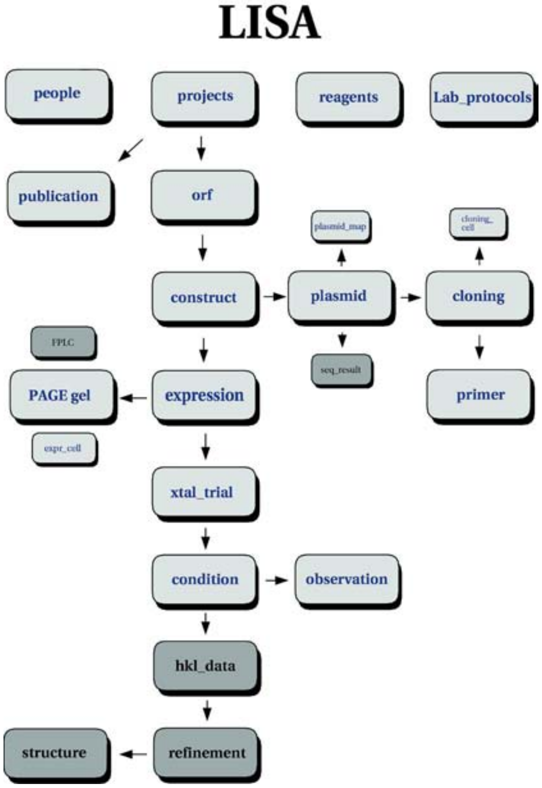


Figure 2
Organization of the LISA database. The top level of the database includes four main categories for tables containing data for personnel and research projects and for laboratory reagents and protocols. Representative links to tables below the projects entry are shown by arrows.

secure web servers can be employed to protect sensitive data when LISA is accessed from insecure networks. Data is accessible to any registered user, but to ensure data integrity only owners of a particular entry have the right to modify or delete it.

Information is stored in tables, where each row corresponds to a record or database entry, *e.g.* data about a particular protein. Each column or cell in that row represents a different attribute of that entry *e.g.* the protein sequence, extinction coefficient, molecular weight *etc*. In relational databases, tables are linked by unique identifiers allowing complex searches across numerous tables. The LISA user interface shows links between different tables as familiar web-browser hyperlinks for easy navigation between related tables (Fig. 1).

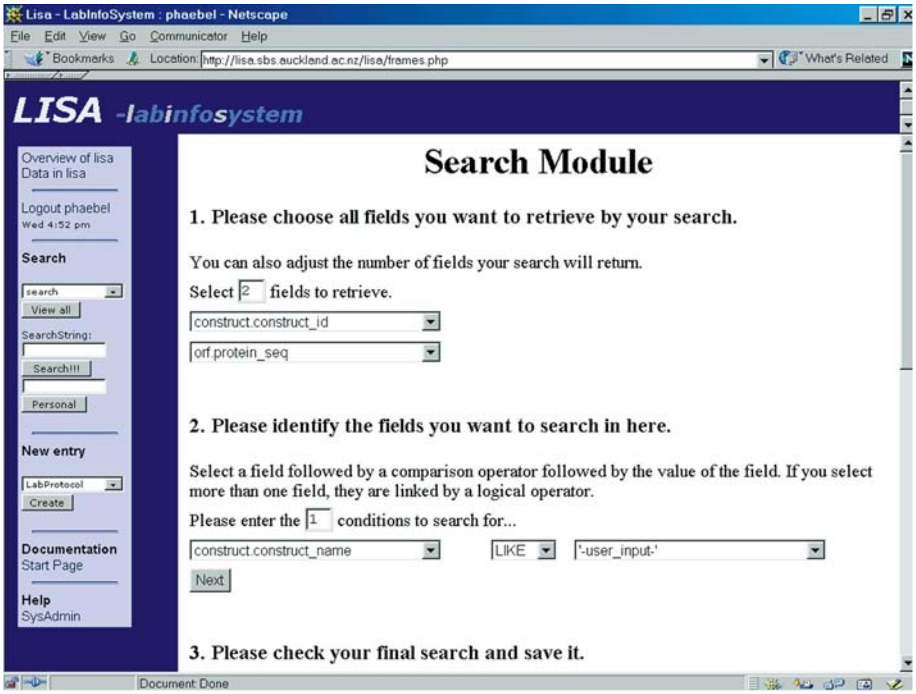
The organization of the LISA database developed for our structural biology laboratory is shown in Fig. 2. The database currently has four top-level categories including information for personnel, research projects, reagents and laboratory protocols (Fig. 2). These main categories contain information such as the as the telephone and address list, descriptions for all the research projects and data about available stocks of chemicals, plasmids, primers and cell lines. Tables also accommodate information about open reading frames, properties of proteins, protein expression, purification and the results of crystallization trials. Images of digitized protein and of DNA gels and PDB coordinate files can be uploaded and stored in the database. LISA is highly customizable and users are encouraged to modify the structure of the database according to their needs. New tables can be readily integrated and existing ones modified without changing the program.

LISA is based on an easily learned graphical user interface derived from the freely available software package *phpMyAdmin* (Ratschiller, 2000). The interface allows users to enter and retrieve data and to carry out various search functions. In addition, advanced searches can be composed and stored in the database. A search module helps users to program new complex searches in the SQL database language (Fig. 3).

Dynamic data cells allow for a high degree of flexibility in data processing. Data cells can be associated with short scripts and programs called modules. Modules control the way data is processed prior to its storage or display on the screen and can be used to automatically generate content from existing data. Examples include the automatic calculation of primer annealing temperatures from the primer sequence or the display of cell contents as interactive hyperlinks.

In addition, modules allow the functional extension of LISA. An example of this is the search module, which enables users to create individualized queries and store them in the database for future use. The search module also helps users write search routines in the SQL database language or upload SQL search routines from files. The file upload function is itself a module that supports the storage of data files uploaded from remote clients (*i.e.* the user's desktop) and their retrieval from the LISA server. Uploaded files are copied to a user-specific directory on the LISA server and a link to the file location is stored in the database. Data import and export functions have been included to enable LISA to exchange information with other databases or widely used office packages.

1342 Haebel *et al.* • LISA database *Acta Cryst.* (2001). D**57**, 1341–1343



Search Module

1. Please choose all fields you want to retrieve by your search.

You can also adjust the number of fields your search will return.

Select 2 fields to retrieve.

construct.construct_id

orf.protein_seq

2. Please identify the fields you want to search in here.

Select a field followed by a comparison operator followed by the value of the field. If you select more than one field, they are linked by a logical operator.

Please enter the 1 conditions to search for...

construct.construct_name LIKE '-user_input-'

3. Please check your final search and save it.

Figure 3
The LISA search module. Modules extend the functionality of LISA by controlling the way data is processed before it is stored in the database or displayed on screen. The search module assists users in formulating complex database searches in SQL. Pull-down menus allow the user to define the search conditions and to choose what information will be retrieved. The compiled search can be tested and stored in the database for future use.

3. Technical aspects

LISA is based entirely on freely available open-source software. Software within LISA is written in *PHP*4 (The PHP Group, 2001), a server-side embedded scripting language (Meloni, 2000; Stoll, 2000). LISA generates interactive HTML pages allowing the remote user to communicate with a central database server *via* standard web browsers. The database software *phpMyAdmin* (Ratschiller, 2000) is used to carry out administrative tasks.

LISA is designed to run on any operating system that provides a *PHP*4 (The PHP Group, 2001) capable web server and a *MySQL* relational database (MySQLAB, 2000). These include most Unix platforms (*i.e.* Linux, Irix, HP-UX *etc.*) and Windows9*x*/2000. LISA has been tested on a PC with SuSE 6.4 Linux distribution (SuSE Inc., 2001) that includes an Apache 1.3.12 web server (Apache Software Foundation, 2001) with *PHP*4 module and *MySQL* 3.22.32 database software.

4. Current use and future development

The development of LISA has allowed us to effectively store and manage data from protein crystallography projects including information from a growing structural genomics programme. The successful use of the system in the past year has demonstrated the feasibility of the central web-based database server together with remote web browsers to manage laboratory information. LISA currently contains database tables and modules for most aspects of cloning, expression and purification of target proteins, including images of DNA and protein gels. Tables for storage of laboratory protocols, an address list, projects and a reagent list have also been implemented.

A module dealing with protein crystallization is currently under development. This module will store information about the progress of crystallization experiments along with images, using a recently published crystallization scoring system for the results (Zeelen, 1999). By recording electronically the results of all crystallization experiments, we intend to design improved crystallization trials. Furthermore, linking all project information in a relational database will enable users to quickly track a project back from the three-dimensional structure to a particular crystallization trial, the protein batch used and the expression and cloning details of the protein.

LISA has been constructed using open-source software packages that have been developed collaboratively by the open-software community (Apache Software Foundation, 2001; MySQLAB, 2000; The PHP Group, 2001). We intend to follow this development model. For that reason, the LISA software is freely available and users are encouraged to contribute to the future development of LISA by adding modules and database tables that deal with new aspects of structural biology, molecular biology and biochemistry.

5. Documentation and availability

A web preview version of LISA 0.9 can be accessed at http://labinfo.sourceforge.net. LISA software for Unix and Windows can be freely downloaded from this site. The Windows distribution contains LISA software plus pre-configured web and database servers. The project development is hosted at the open-source software development site Sourceforge (http://sourceforge.net/projects/labinfo), which also provides a support forum. Users are encouraged to customize the database and to write their own modules to extend functionality.

This research has been supported by the Health Research Council of New Zealand and by a Major Equipment Grant from the Wellcome Trust. PWH gratefully acknowledges support from Boehringer-Ingelheim Fonds.

References

Apache Software Foundation (2001). *Apache Project*, http://httpd.apache.org.
Berman, H. M., Bhat, T. N., Bourne, P. E., Feng, Z., Gilliland, G., Weissig, H. & Westbrook, J. (2000). *Nature Struct. Biol.* **7**, 957–959.
Goodman, N., Rozen, S., Stein, L. D. & Smith, A. G. (1998). *Bioinformatics*, **14**, 562–574.
Jakobovits, R., Soderland, S. G., Taira, R. K. & Brinkley, J. F. (2000). *Proc. AMIA Symp.*, pp. 374–378.
Meloni, J. C. (2000). *PHP Essentials.* Roseville: Prima Tech.
MySQLAB (2000). *MySQL*, http://www.mysql.com.
The PHP Group (2001). *PHP: Hypertext Preprocessor*, http://www.php.net.
Ratschiller, T. (2000). *phpWizard – Building Dynamic Websites with PHP*, http://www.phpwizard.net.
Stoll, R. D. & Leierer, G. A. (2000). *PHP4 + MySQL*, 1st ed. Duesseldorf: Data Becker. In German.
SuSE Inc. (2001). *SuSE Inc.*, http://www.suse.com.
Thornton, J. (2001). *Trends Biol. Sci.* **26**, 88–89.
Zeelen, P. Z. (1999). *Protein Crystallization*, edited by T. M. Bergfors, pp. 133–141. La Jolla: International University Line.