CABIOS

Vol 10 no.2. 1994
Pages 137 – 143

A distributed system for two-dimensional gel analysis

P.J.Monardo, T.Boutell, J.I.Garrels and G.I.Latter[1]

Abstract

The Quest II system is a new two-dimensional (2D) gel analysis software system for the construction and analysis of 2D gel protein databases. A new architectural approach to 2D gel software systems has been utilized. This architecture is based on a tightly coupled client/server model. There are three layers to the system architecture: (i) a database layer consisting of three database servers, (ii) a compute layer consisting of three compute servers and (iii) an extensible user interface layer currently consisting of user interface tools for linearization and merging of scanned images, the segmentation and detection of protein spots on the images, matching, editing, and analysis of gels. The ability to store and retrieve the large volume of spot data inherent in 2D gel analysis while utilizing database technology is demonstrated.

Introduction

In this paper we describe the Quest II system architecture which represents a new architectural approach to the QUEST system (Garrels, 1989), a two-dimensional (2D) gel analysis system. The new system architecture is designed to meet the growing needs of the QUEST Protein Database Center, to be deliverable to protein database builders, and to take advantage of a modern networked computing environment.

The QUEST Protein Database Center

The Quest II system was developed at the QUEST Protein Database Center, a NIH Biomedical Research Technology Program center dedicated to the development and use of 2D gel electrophoresis technology. The QUEST Center consists of a computer facility where the gels are analyzed and protein databases are built. The goal of the Center is the construction of protein databases for scientific investigation. The QUEST center hardware currently consists of a central data and compute server, a development server, a cluster of four workstations used for program development and data processing, and an MD 300A laser densitometer and a shared Fuji BAS2000 phosphorimager.

Cold Spring Harbor Laboratory, PO Box 100, 1 Bungtown Road, Cold Spring Harbor, NY 11768, USA

[1]To whom correspondence should be addressed

As a center for technology development and distribution, we needed to develop technology to create 2D gel databases and to make these databases and the technology more readily available to the wider scientific community. Identifications of proteins in 2D gels is beginning to be made available in static ways (periodic updates in a range of quarterly to annually) via journal publications such as the annual database issue of *Electrophoresis* and via the Internet with the publication of the ECO2DBASE (Van Bogelen *et al.*, 1992) via the National Center for Biotechnology Information repository. We needed to develop database technology that would allow us to store annotations and quantitative information for hundreds of gels. These data need to be made available dynamically, first within a single institution, and ultimately via the Internet. Central resources for database construction become more practical in view of the growth in speed and availability of networks compared with the expense of accumulating the many basic experiments and identifications necessary to construct a large quantitative database. Our solution is to construct a technology which can be provided to sites which will be building databases and that can make these databases available via networks.

In this paper we will provide a background description of 2D gel analysis and the requirements of a new software system. Our architectural solution to these requirements will then be described.

The 2D gel analysis system

The basic algorithms of the system detect, quantify and match the spots on gel images. After detecting spots from single gels, these spots are matched to spots detected from other gels. A set of matched gels is referred to as a 'matchset'. Analysis is performed on a matchset. Quantitative variation of spots can be traced across a matchset, or additionally across matchsets which contain what we refer to as 'linker gels', i.e. gels which appear in multiple matchsets (Garrels and Franza, 1989) . We store all of the above information in a database which also contains spot annotative information. The data is stored as parameterized models of the spots (Gaussians in our case), and this is a key factor in our ability to view the data over a network at reasonable speeds.

In a previous paper, Garrels (1989) described in detail the steps involved in the analysis of 2D gels. A brief summary of the data flow is given below:

Image acquisition and merging from multiple film exposures. Digitization of radiolabeled 2D gel films occurs after multiple exposures to film or phosphor imaging plates. To increase the dynamic range of the data, these multiple exposures can be merged together using calibration data into a single 16-bit image after linearization of the separate images.

Background subtraction and detection. Gels contain vertical streaks and diffuse haze which must be removed, and the spot centers must be located. The background subtraction and detection algorithms proved adequate and robust for the gels produced in the CSHL 2D Gel Laboratory Core Facility (Garrels, 1989). Improvements to these algorithms have been made in this system and will be be described in detail elsewhere.

Segmentation of the image into 2D Gaussian spot models. The spot centers are located, and 2D Gaussians are then fitted to the spots. A single spot may be modeled by several Gaussians. The Gaussians are represented by five parameters consisting of two half-widths, the x and y coordinates of the center of the Gaussian, and the height of the Gaussian. The raw images no longer need be kept on-line once the spot model representation of the image has been created and edited to satisfaction. This use of the 2D Gaussian spot model allows us to reconstruct a model of the gel from the spot database. The spot model therefore serves as a compression technique, as well as a method of quantifying and analyzing the 2D gel protein spots. A comparison of a synthetic image (the calculated image obtained from the 2D Gaussian formula and the Gaussian parameters of the spots) to a bitmapped image was shown by Garrels (1989).

Editing: The QUEST system has a number of editing facilities for modification of results obtained by automatic features of the system or to make corrections based on new information obtained from, for instance, new experiments.
Combining: Combining spots is the operation of grouping multiple Gaussians together to represent a single protein spot. Spots may be uncombined as well.
Canceling: Canceling a spot refers to the operation of tagging a spot so that it will not be considered a detected protein. This operation is used, for instance, when portions of background are detected as spots.
Adding: A spot may be added to a region where the user observes a spot which was not detected, or spots may be automatically added at positions where they appear in other gels in the matchset (but not in the current gel). This assists the matching process. After adding a spot, the point at which the spot is added is considered the spot center and the spot can be refitted, based upon the gel image.

Matching. A matchset is a group of gels organized together as an experiment. Corresponding spots on separate gels of a matchset are matched to each other for analysis. There are four phases to matching: (i) landmarking, in which an operator chooses a spot on one gel and indicates the matching spot on each of the other gels; (ii) propagation of matches, in which matches are extended to nearby spots if they are within a specified distance and direction; (iii) neighbor matching, in which new matches are found by using matches in a neighborhood to compute local transformations; and (iv) crossmatch, in which consistency is checked by examining all matching pairs at one time. Once sets of spots are matched, data can be exported for analysis or analyzed using the analysis features of this system.

The previous version of the QUEST system was used to construct the REF52 database (Garrels and Franza, 1989) and a mouse embryo database (Latham *et al.*, 1992). The REF52 database proved useful for a study of the transformation-sensitive and growth-related changes of protein synthesis in REF52 cells. The published database contained 79 gels.

We undertook an analysis of the requirements of a system that could meet the growing needs of our resource center and would allow us to provide our software to other groups interested in the construction of 2D gel protein databases.

Implementation

Requirements of this new 2D gel database system

Requirements analysis revealed that it was necessary for the system to meet the following criteria:

Performance of spot record loading. We typically load of the order of 3000 spots per gel from 4–10 gels. The underlying database system needs to be able to perform the loading of these 12 000–30 000 spots in ~1 min.

Scalability. The ideal architecture would be scalable and would take advantage of a range of hardware from a single processor workstation to a center such as QUEST, consisting of multiprocessor central servers surrounded by single or multiprocessor desktop workstations.

Cost. We need to keep the cost low so that we could provide this technology to individual scientists working on single desktop workstations who may not be able to afford the resources necessary to purchase and maintain a full database management system.

Separate the user interface from the remainder of the programs. Separating the user interface programs from the remainder of the software allows other user interfaces to be adopted more easily if necessary.

Separation allows multiple separate programs (referred to as tools) to provide separate views of the data, and to be used simultaneously.

Keeping the user interface and graphics in a separate process allows us to offload this processing to the desktop workstation for multiple machine workgroups served by central compute servers.

Extensibility: we wanted the user interface to be extensible, i.e not limiting the tools to our initial set but rather allowing many views of the data.

Ability to offload intensive computing from the desktop workstation. Several aspects of 2D gel analysis utilize considerable resources and are computing-intensive enough to be considered batch processes. These processes are best separated from the remainder of the code so that the user can continue with interactive aspects of their 2D gel work separately from the computing-intensive parts. This will allow us to take advantage of multiprocessor machines as well.

Dynamic data analysis and editing. The nature of current 2D gel production technology requires a dynamic view of the data, i.e. the ability to reinterpret 2D gels based on new experiments is necessary. The ability to edit databases directly during analysis is therefore an important feature.

Take advantage of high end database technology with little porting effort. We wanted to keep the option available to move to full database management systems if ever their performance became adequate and their cost dropped to a point that would allow single scientists to have access to this on their desktop, or if high-end users had machines, disks and database management systems whose performance was adequate.

Network accessible data. We felt that it was necessary to have the databases dynamically accessible over a network, allowing a single database to be built by multiple collaborators over a local area network (LAN). Particularly at our center, there are users in separate buildings who are not on our same NIS (network information system) domain, but are on our LAN. We would like to ultimately provide dynamic access to 2D gel data via the Internet.

Architectural solution

We chose to implement a client/server architecture utilizing ISAM database technology and interface tools constructed using XView to construct OpenLook compliant interfaces. The new architecture consists of a database layer with three database servers, a compute server layer with three compute servers, and user interface tools to communicate to these servers. This is presented in Figure 1.

There are currently two user interface tools used to communicate with these servers: the QuantTool, which processes data from import of raw pixel data through the viewing and acceptance of the Gaussian spot-modeled image; and MatchTool, a matching and editing tool. These interface

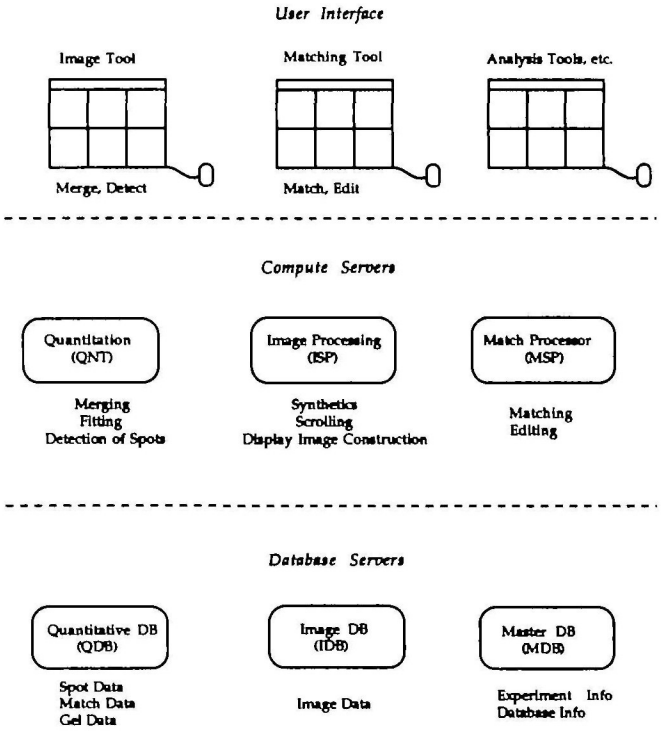

Fig. 1. QUEST II system architecture. The three layers of the Quest II architecture. The two user interface tools currently in use, QuantTool and MatchTool, are shown, and the extensibility of the user interface will allow additional tools to be added.

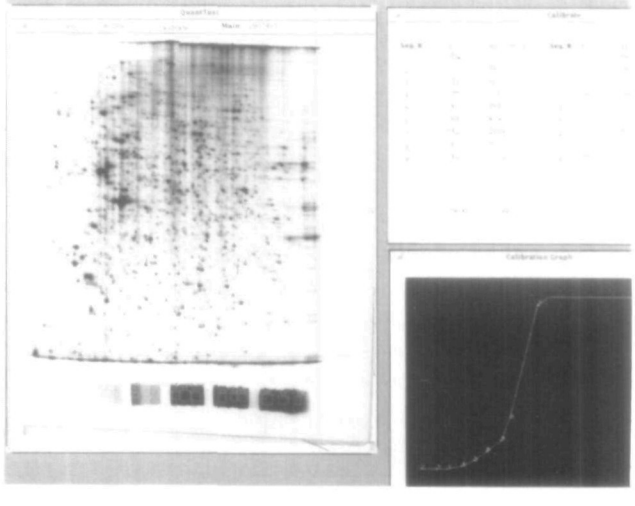

Fig. 2. QuantTool. Utilizing this tool the user examines the raw data, places the cropping box to delineate the region of analysis, and marks off the segments of the calibration strip used in linearization of the data. After examining a set of raw scans, this tool is used to send the data off for linearization, merging, background subtraction and spot detection. This screen dump shows a crop box placed on a 2D gel image with boxes placed on the calibration strips.

tools are shown in Figures 2 and 3. The number of user interface tools as well as the number of servers may grow under this scalable architecture. While some analysis capabilities are built

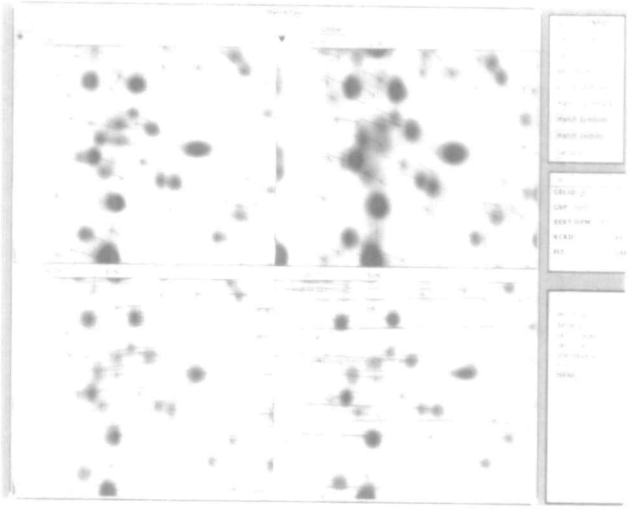

Fig. 3. MatchTool. With this tool the user matches and performs preliminary analysis of sets of gel images called matchsets. Editing, landmarking of initial matches and automatic matching are performed with this tool. Quick flickering between synthetic images and gel-based images is provided in this tool (as it is in the Imageset tool). This allows the user to compare the synthetic image constructed from the Gaussian models to the original gel-based image.

into the current matching and editing tools, we are planning full analysis tools.

Databases

As shown in Figure 1, the database layer consists of three database types: the master database, the quantitative database and the image database.

Master database (MDB). The master database contains experiment information, host locations for the other databases, and account information. There is only one master database for a LAN, because it is used to keep track of the location of the other databases. There will tyically be multiple quantitative and image databases on different host computers.

The tables in the master database are: the database table (host and location of the other databases); the account table (user account information); the experiment table (information on the experiment being performed); the sample table (data on sample used in gel production); the gel table (information about the gels and their production); and the exposure table (information about the exposure of the radiolabeled gel).

Image database (IDB). The image databases contain the digital images, consisting of raw images of digitized 2D gel data, images obtained from the processing of this raw data, and temporary images such as synthetic images generated from the segmented spot data. The raw images and processed images are eventually moved onto tape from disk and their tape locations are maintained in the image database. The synthetic images are of a temporary nature and exist for display purposes only.

The tables in the image database are: the imageset table (information about sets of images); and the image table (information about individual images; pointers to the images).

Quantitative database (QDB). This database contains the spot based data, the quantitative and annotative data, and the matching information.

The tables in the quantitative database are: the gel table (gel quantitative information such as the number of spots and the scale factor relating gel data to d.p.m.); the spot table (size, location and other information on spots); the spotset table (number of spots in spotsets and names of spotsets); the spotlist table (lists of spots referred to by entries in the spotsets table); the match table (the table of correspondence pairs of matching spot IDs); the matchlist table (table of correspondence pairs of matched gels); the info table (information table containing up to 8K of text and a reference to one of the basic objects in the system—gel, spot, spotset—as well as a reference to an attribute); the attribute table (table of attributes which can be referenced by entries in the info table).

Compute servers

There are three basic compute servers, one to perform the display image processing tasks, one to perform the quantitative tasks and one to perform matching.

Imageset protocol (ISP). The ISP service creates displayable images. Displayable images are created from raw and processed bitmap images and also synthesized from Gaussian spot data. Since images are memory mapped, the ISP service provides a pointer to these mapped images for display. The ISP service may reside on the local machine where the image is being displayed, or on a remote machine.

Matchset protocol (MSP). The MSP service is responsible for matching of gel data and communications with the MatchTool. The MSP service maintains an in-core representation of the matchlist in a multiply linked data structure. An example of a communication between the MSP service and the MatchTool is a request from the MatchTool to MSP for the locations of the centers of spots in a given rectangle, with the returned response from MSP being a point list of the centers of the spots in that rectangular region in real world coordinates, which MatchTool can then display utilizing its graphics display libraries. The MSP service may run on the displaying workstation, or may run on a separate machine.

Detection and merging service (QNT). The QNT service performs tasks such as multiple exposure merging, image linearization, spot detection and fitting two-dimensional Gaussians to these spots. There is a QNT_MGR service which manages the task queue of requests to QNT and sets the time during which QNT tasks will run on a particular host. Processing by the QNT service is compute intensive and may

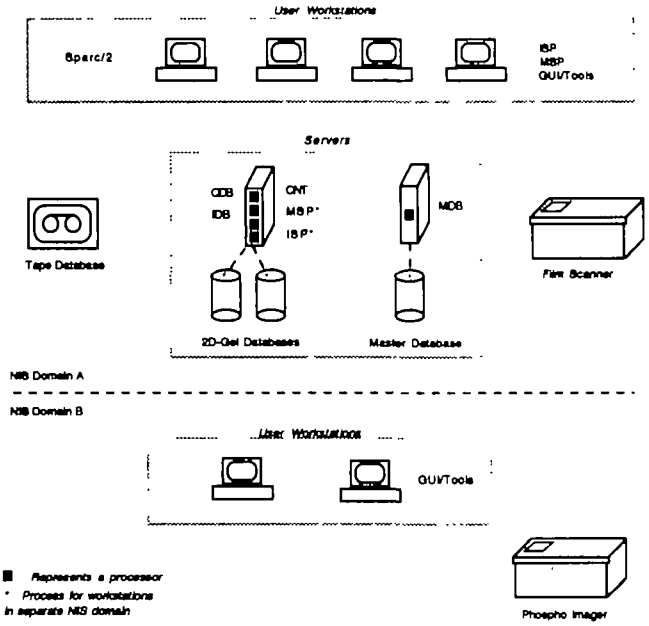

Fig. 4. Resource utilization at the CSHL QUEST Protein Database Center. At QUEST we utilize a multiprocessing capable Sun SPARCstation 10/42 for intensive computing. The QNT server, used for batch-oriented computing on pixel-based images, resides on this machine. Note also that the image display processing (ISP) for users in other buildings at CSHL (not in our NIS domain) also occurs on the multiprocessor machine. Their user interface tool and their graphics display processing occur locally on the machine on which they are viewing the image so that interactive graphics need not be transmitted over the network.

be run on a different machine from the display machine, or may be run at a time when the display machine will not be in use.

All servers in the system can run on a single CPU, but greater advantage of the architecture is taken by running servers on multiple machines or multiprocessor machines. Figure 4 shows the hardware configuration at the QUEST protein database center and how we typically configure our services and hardware. The architecture takes advantage of multiprocessor machines. For example, the QuantTool could be running on a graphics workstation while the processing of these images is performed on a separate multiprocessor machine, with each merge or detection utilizing a different quantitation service (QNT) on a different processor on that machine.

User tools

QuantTool. QuantTool is the user interface tool employed to process data from raw data import through detection and fitting of spots. It is the primary interface to the QNT quantitation service. An example of the QuantTool in use is shown in Figure 2.

MatchTool. The MatchTool (Figure 3) provides the interface to the matching process. It allows the display of all images in a matchset simultaneously and provides user interface to assist in matching and editing. With the MatchTool the user sets landmarks for matching, then after running automatic matching, views and edits the results. There are some basic analysis features such as checking the variation of integrated d.p.m. of spots across a matchset and creation of spot sets.

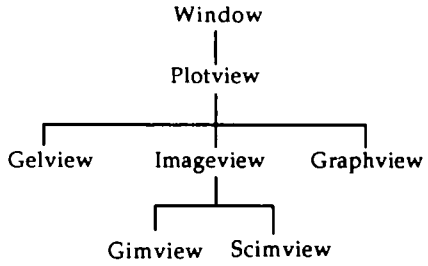

Fig. 5. XView class hierarchy example. An example of an XView object-oriented class hierarchy used in our system. These classes (or packages in XView terminology) are all subclasses of window, and are visual objects. All of these objects are subclasses of plotview. Gelview objects are the visual window into gels in the MatchTool, the imageview objects are the basic view of our data in the QuantTool, further subclassed to provide the ability to view scans (scimview) or gels (gimview). Graphviews are utilized as part of our analysis tools.

System and methods

Technologies employed

XView. The user interface code was built using XView (Sun Microsystems Inc.). XView allows construction of new classes of objects which can inherit from standard XView objects. Figure 5 shows an example of an inheritance tree for a set of objects we have created. This view tree inherits from the standard XView window class; other XView objects we have created inherit from the XView generic class.

RPC. RPC (Sun Microsystems Inc.) was chosen as the client/server communication method. This is a well-established technology and has proven to be robust for our system.

XGL. XGL (Sun Microsystems Inc.) was chosen as our graphics library for display of graphics overlays on our images. We chose XGL because of its speed; XGL utilizes DGA, Direct Graphics Access, which bypasses the X-Windows server overhead when displaying on the local machine. XGL also provides advanced rendering and three-dimensional capabilities which we intend to employ in later versions.

OpenWindows Version 3.0/XView. OpenWindows Version 3.0/XView (Sun Microsystems Inc.) was chosen as our windowing system and interface toolkit because of its functionality and availability on Sun workstations.

Net ISAM. Net ISAM (Indexed Sequential Access Method) v. 1.0 (Sun Microsystems Inc.) was chosen as our database access

method because of its speed, low maintenance overhead and low cost. This choice is described in detail below.

Memory mapping. All images are memory mapped which allows the system to handle large data volumes. A typical imageset contains ten 8 Mbyte processed images and up to forty 10 Mbyte images of raw data.

Utilize database technology

We needed to utilize database technologies in order to build large, robust databases of 2D gel spots. All spots for a particular 2D gel database need to exist in a single database file with a fast access method which provides a structured view of our data. However, we did not need the full functionality of a relational or object-oriented database management system, and could not use such a system because both performance problems and the cost and maintenance overhead would make it difficult to provide this to other centers who may not have such a fully supported database management system. We typically need to retrieve 3000 or more spots from each of 10 or more gels in <1 min. Our test with a standard relational database management system (Sybase) showed that the retrieval of large numbers of records at one time was not fast enough for our needs. Such systems are typically designed for fast retrieval of a single record from a large number of records, whereas we need to retrieve very large numbers of records at once. By using a record management system we have added structure to our data and gain fast searching available via B* trees, which are well-balanced balanced binary search trees that provide very fast access.

Although the NetISAM software provides the ability to access data over the network, the performance was not sufficient. Every record requested constituted a separate system call and a separate network packet with their associated overheads. We constructed the QDB server so that all records are retrieved from the database machine utilizing ISAM and are then grouped together into a single network packet which is passed to the requesting process.

Network access to data

Because of the nature of our center, we would like to have the ability to allow access to our data via the Internet. Our new architecture will allow scientists to access our data via the Internet and to use analysis tools to view the data locally at their site. This is a more dynamic model which will allow scientists to access the latest changes to a database. This could be done in several ways, the most practical being to import remotely the compressed spot record data and then perform the viewing and displaying of images locally at the remote site. A local MSP service initiates the request to the QDB database at the remote site for the spot data for the matchset being viewed or analyzed. This spot record data is then transmitted as described above to the remote site and is stored in core in the MSP service. Display of the synthetic images is then performed using a local IDB so that the display images are not transmitted over the network.

Discussion

Solution

Our requirements were met by this client/server architectural approach.

- Spot loading and display performance is satisfactory. In our configuration the master database resides on a Sun SPARCstation 4/330, the display takes place on a SPARCstation 4/65 and the spot image generation takes place on a SPARCstation 10/42. In this configuration it takes 15 s to load spots and display the image data for a four gel matchset with ~3000 spots per gel.
- The system is scalable. The system can run on a single SPARCstation 4/65 class machine, and we have run the system here on configurations taking advantage of multiple CPUs as described in the previous paragraph.
- We have been able to keep the cost low enough to make it feasible to provide this service to other scientists. The cost of ISAM technology is more than an order of magnitude less than that of a relational database.
- The user interface tools have been separated from the quantitation services and the databases as can be seen from the architecture shown in Figure 1. We have already constructed separate tools with different views of the databases as demonstrated in this paper by QuantTool and MatchTool.
- Intensive computing can be offloaded for users with multiple CPUs. This is clearly another benefit of our architectural approach and was illustrated in the example described previously in this section which used three separate CPUs. A typical mode of operation here is to have the main intensive computing (QNT) take place on a Sun SPARCstation 10/42 managed by the queue manager (QNT_MGR), while the user prepares additional images for quantitation on his/her desktop workstation. Images are locked while being processed so that another user is prevented from viewing them, by taking advantage of a simple high-level locking server running on the same CPU as the master database.
- Our choice of ISAM for a unified spot storage database has proven successful. It gives us the ability to access large numbers of spot records rapidly. We can edit spot records and perform matching of a matchset, and then later analyze them as part of the same database rather than reloading data into a separate database system for analysis.
- We could port sections of this system to a high-end DBMS, except for QDB, because the spot record transaction time is not fast enough with relational

systems. Our database library functions were modeled on the style of those in standard relational packages, so that we could incorporate a true relational database without a major rework of the higher level layers.

- RPC gives network access to the data. We are currently experimenting with the ability to view synthetic images outside of our LAN. Preliminary data indicate that this will be feasible. The accessing and transport time for spot records seemed sufficient.

Another advantage of this architecture is that because the modules run as separate processes they exist in separate address spaces.

Future enhancements

As stated earlier, initial experiments have indicated that we will be able to allow access to our synthetic images via the Internet for users with our client software running on their local machine. We will be doing further experimentation and development in this area.

We will be emphasizing development of analysis tools. Researchers should be able to analyze 2D gel databases on several levels. Analysis of a single matchset (or experiment) is the most common. Analysis across matchsets of the data in a single database can be done with our network database architecture as long as there are linker gels between the matchsets across which analysis is desired. Cross-database analysis will also be explored in the future. We believe that our architecture is well suited to this. Separate databases can reside on separate computers on separate filesystems, and could still be cross-analyzed if the appropriate cross-database linkages are in place.

The Quest II software is available to users at non-profit institutions who are interested in building large databases. Interested individuals should contact the corresponding author (latter@cshl.org).

Acknowledgements

We would like to acknowledge the continuing support of Dr James Watson, and we would like to thank Bob Franza and Scott Patterson for their support, patience and advice during the construction of this system. This work has been funded under the NIH-DRR (grant no. P41RR02188).

References

Garrels,J I. (1989) The QUEST system for quantitative analysis of two-dimensional gels. *J. Biol. Chem.*, **264**, 5269–5282.

Garrels,J.I. and Franza,B.R. (1989) The REF52 protein database *J. Biol. Chem.*, **264**, 5283–5298.

Garrels,J.I., Franza,B.R., Chang,C and Latter,G I (1990) Quantitative exploration of the REF52 protein database. cluster analysis reveals the major protein expression profiles in responses to growth regulation, serum stimulation, and viral transformation. *Electrophoresis*, **11**, 1114–1130.

Goodman,A.M., Burke,P., Funk,M., Vargas,R.J., Snydr-Michal,J.T., Appel,R.D. and Hochstrasser,D.F. (1991) A model for a distributed database to manage large 2D-PAGE collaborative experiments. In Dunn,M.J. (ed.), *2D-PAGE '91*. National Heart and Lung Institute, London, pp. 7–77.

Latham,K.E., Garrels,J.I., Chang,C. and Solter,D. (1991) Quantitative analysis of protein synthesis in mouse embryos. 1. Extensive reprogramming at the one and two cell stages. *Development*, **112**, 921–932.

Van Bogelen,R.A., Sankar,P., Clark,R.L., Bogan,J.A. and Neidhardt,F.C. (1992) The gene-protein database of *Escherichia coli*. *Electrophoresis*, **13**, 1014–1054.

Young,D.A. (1984) Advantage of separations on 'giant' two-dimensional gels for detection of physiologically relevant changes in the expression of protein gene-products. *Clin. Chem.*, **30**, 2104–2108.