The Art of Molecular Graphics

The 1998 Molecular Graphics Art Show

David S. Goodsell, The Scripps Research Institute, Department of Molecular Biology, 10550 N. Torrey Pines Road, La Jolla, CA 92037, *goodsell@scripps.edu*

Once every couple of months, I am overcome by the need to forget about my research, put off writing that grant proposal until tomorrow, close the office door, and then to sit down and focus my attention completely on creation of an image. I trade the world of conformations and coordinates for the world of color harmony and composition. This desire obviously strikes many researchers, as revealed in the recent Molecular Graphics Art Show, held in conjunction with the 17th meeting of the Molecular Graphics and Modelling Society.

Seventeen individuals provided artwork for the show. Their professional backgrounds are different, but they share an interest in molecular subjects. Some are scientists, taking a break from their research to explore their artistic side. Some are scientific illustrators, creating artwork to teach molecular science. Some are artists who find their inspiration in molecular science. All have created imagery that compellingly depicts the invisible molecular world. The entire show, and the previous 1994 show, may be viewed currently on the World Wide Web at *http://www.scripps.edu/pub/goodsell/mgs_art*. A few examples, taken from members of the society, are included here.

In "Fumagillin" (Figure 1, page 56), Richard Gillilan of the Cornell Theory Center uses color and lighting to emphasize the important features of molecular interaction. The fungal metabolite fumagillin forms a covalent bond with a histidine in the enzyme methionine aminopeptidase 2, blocking the active site and leading to the inhibition of blood vessel growth. "The new bond," he writes of the image, "still hot from the energy of formation, illuminates the space while the tail of the inhibitor casts its shadow on the rear wall of the binding pocket."

Wm. Michael Carson of the University of Alabama at Birmingham provided an allegory on the process of macromolecular crystallography. In "Diffraction Space: The Final Frontier" (Figure 2), area detector data becomes a landscape over which an insulin molecule rises. Carson uses images such as this to test the abilities of new software, in this case, the IRIS Explorer from Silicon Graphics, Inc.

"Nanoscape I: Encounter in the Blood Stream" (Figure 3) by Arthur J. Olson of the Scripps Research Institute combines a number of graphical methods within the AVS dataflow environment. The image depicts an illustrative volume of blood plasma. Over 500 protein subunits, representing over 1.5 million atoms, are depicted using a spherical harmonics approximation of the protein surface. An invading virus, in green, is attacked by antibodies, in pink and light blue, within a field of other proteins, in shades of blue. The image was designed to be shown stereoscopically and a 3D model was created. "Thus," Olson writes, "the piece has sculptural as well as pictorial elements."

Teresa A. Larsen, juror for the show, transformed a virtual sculpture into a physical sculpture in "HIV" (Figure 4). A model of the virus at 1,000,000 times magnification was constructed using a laminated-object, rapid prototyping device, based on a 3D model created for a previous animated short film. The model, which may be disassembled to reveal the inner structure of the virus, serves as a prototype for manufacture of handheld plastic models for education.

News and Views

INDEX

Journal of Molecular Graphics and Modelling

655 Avenue of the Americas, New York, NY 10010

1093-3263/98/$19.00

ELSEVIER

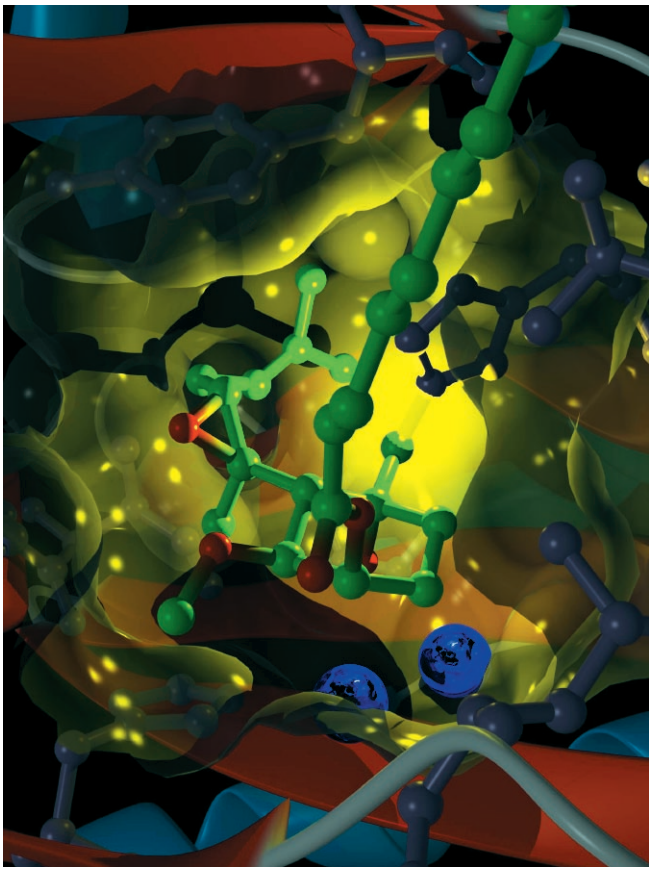

Figure 1. Fumagillin by Richard Gillilan

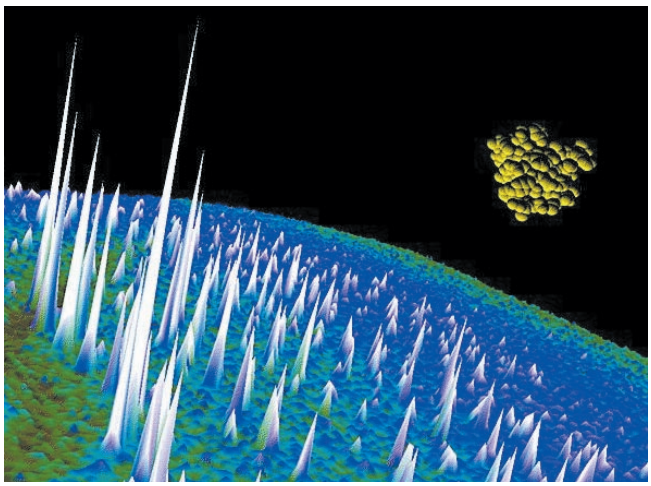

Figure 2. Diffraction Space: The Final Frontier by Michael Carson

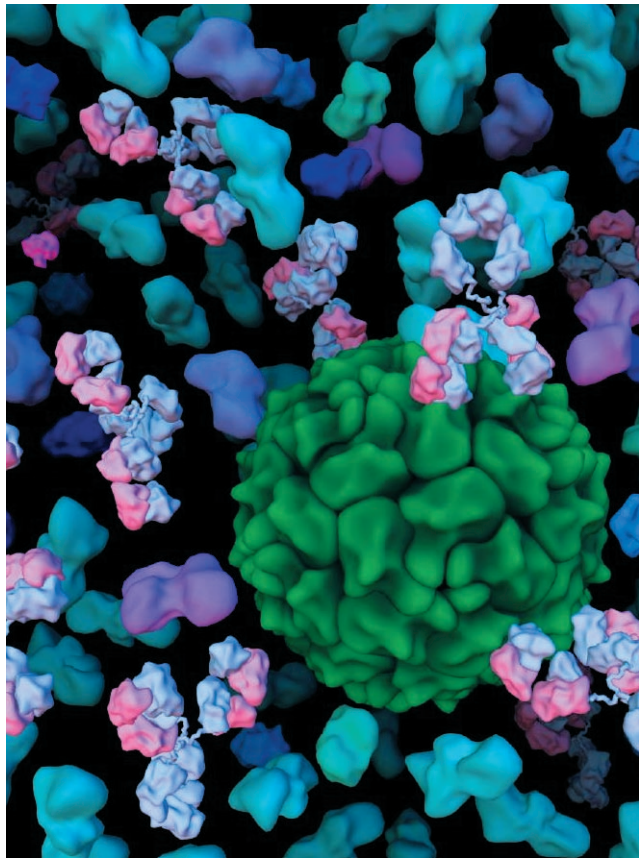

Figure 3. Nanoscape I: Encounter in the Blood Stream by Arthur J. Olson

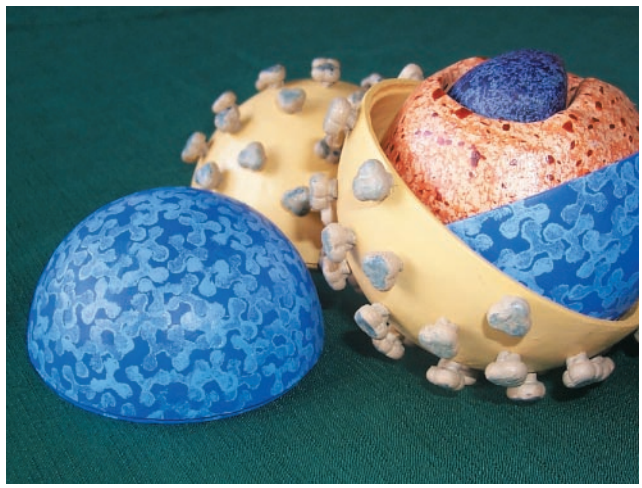

Figure 4. HIV by Teresa A. Larsen

Molecular biology has been a fertile ground for the collaboration of art and science, perhaps rivaled only by the explosion of natural history illustrations created as the taxonomy of plants and animals first fell under scientific scrutiny. The reason is clear: whether artist, illustrator, or scientist, a picture is the best way to capture the beautifully intricate, invisibly tiny world of molecules.

Python: A Programming Language for Software Integration and Development

M.F. Sanner, The Scripps Research Institute, 10550 North Torrey Pines Road, La Jolla, CA 92037, sanner@scripps.edu

Over the last decade we have witnessed the emergence of technologies such as libraries, Object Orientation, software architecture, and visual programming. The common goal of these technologies is to achieve software reuse. Although many significant advances have been made in areas such as library design, domain analysis, metric of reuse, and organization for reuse, there are still unresolved problems such as component interoperability and framework design (1). We have investigated the use of interpreted languages to create a programmable, dynamic environment in which components can be tied together at a high level. This work has demonstrated the benefits of such an approach and the features of the interpreted language that are key to successful component integration.

The Problem

One of the challenges in bio-computing is to enable the efficient use and inter-operation of a wide variety of rapidly evolving computational methods to simulate, analyze, and understand the complex properties and interactions of molecular systems. In our laboratory, we investigated several areas including protein-ligand docking, protein-protein docking, and complex molecular assemblies. Over the years, we have developed a number of computational tools such as molecular surfaces, phenomenological potentials, and various docking and visualization programs that we use in conjunction with programs developed by others. The number of programs available to compute molecular properties and/or simulate molecular interactions (e.g., molecular dynamics, conformational analysis, quantum mechanics, distance geometry, docking methods, and *ab-initio* methods) is large and growing rapidly. Moreover, these programs come in many flavors and variations, using different force fields, search techniques, and algorithmic details (e.g., continuous space vs. discrete; Cartesian vs. torsional). Each variation presents its own characteristic set of advantages and limitations. These programs also tend to evolve rapidly and not are usually written as components, making it hard to get them to work together.

The "Traditional" Solution

Typically, researchers have been using tools such as AWK and shell scripts to make such programs work together. Although that approach appears tempting initially, it has many inherent problems and limitations that will surface in the end. These include a very low level of inter-operability, i.e., usually data is transferred between programs using files or pipes allowing only inter-operation at the program level rather than the function level or at least functionality level. This makes its difficult, for instance, for a molecular dynamics code to use some third party electrostatic or molecular surface calculation package to derive a term used to drive the simulation; to use someone's visualization program tools to steer the simulation; or to monitor or play back trajectories. Such developments usually require substantial coding, and understanding of the source code. This approach also requires the creation of a large number of interfaces between different tools, which makes it very hard to incorporate new methods into the tool set and therefore stifles the researcher's creativity. The level of code reuse offered by this approach is very low. For instance, every program operating on molecules will need to implement its own parser for different molecular file formats, each having its own bugs and weaknesses, and each requiring coding effort. Finally, this approach often leads to very large scripts that are difficult to maintain, extend, and debug.

Other Solutions: Visual Programming, Specialized Software Suites

These frustrations have prompted us to investigate better methods for developing code and integrating computational methods. Our first approach was based on using AVS (Advanced Visualization System from AVS Inc.). This environment has proven very useful for us over the last ten years, in terms of code reuse and capturing developments done by a set of transient collaborators (typically post-doctoral fellows who spend a few years in the laboratory). We have also encountered some limitations. AVS is a data-flow driven computation and visualization environment that comes with a large number of processing modules for a wide variety of operations, such as data input, image processing, surface and volume rendering, etc. These modules can be linked together graphically using a network editor to create a processing stream for a particular visualization or computation. It also offers a mechanism for adding custom-designed modules for new computational methods. AVS users roughly fall into three classes distributed in a pyramid: at the high end is the module programmer, typically writing C programs and making this code available as AVS modules. The second, and larger class of users, are those who produce their own networks using existing modules. Although networks do not have constructs for loops or conditional execution, many visualizations can be done at this level without writing a single line of code. The third, and largest class of users, are those who use their own data with an existing network. One of the reasons AVS has worked well for us is probably due to the visual programming paradigm creating this intermediate class of users. AVS is useful for scientists in need of custom visualization, but who do not want to become programmers. Of course, AVS' modular nature promotes code reuse that leads to rapid prototyping. This has enabled the scientist to concentrate on the visualization process rather than the program used to visualize the data.

However, molecular modelling and biomolecular visualization pose many challenging problems for data-flow environments. Molecules have a high level of internal organization which is often desirable to reproduce in the programs operating on them. This is not always compatible with the simple data-types typically available in these environments. There are also problems of data duplication and inter-module communication. However, what we felt was the most restraining limitation in AVS was its lack of scripting capabilities. The AVS Command Language Interface (CLI) merely consists of a set of commands, and it exposes only a subset of the kernel's functionality, creating some serious limitations. We have solved some of these problems by embedding a Python interpreter in an AVS module, thus adding scripting capability (1).

Commercial and academic molecular modelling packages address these problems more specifically than AVS. However, most of these packages are monolithic programs, providing only a limited set of options for altering the style of the visualization, or extending the program to accept new types of data, or to do new computations. Since one of our missions is to investigate new computational methods and visualizations, they did not appear to be the right tools.

Language-centric Approach and Interpreted Languages

Programs are usually developed in a "self-centric" way, meaning that they are written to be self-contained units aimed at solving a given problem or fulfilling a given task. Some programs, like AVS, are designed to be extended by adding new modules that can encapsulate new computational methods, but this always has to be done within the framework of the program. Moreover, since programs are inherently specialized, this is bound to create problems. To address this problem, we decided to experiment with a "language-centric" approach. We use a high-level language as the core of our framework. Rather than writing programs, we now extend this language with modules or components implementing specific functionality. The high-level language serves as a "glue" to tie modules and components together to rapidly create specialized applications. In some sense, the language becomes a "scripting framework" allowing fast prototyping of new applications. Developing extension modules for the language corresponds to postponing specialization of code as much as possible.

We felt that an interpreted language would provide the flexibility, interactivity, and extensibility needed for such an approach, and we started exploring using the three most popular interpreted languages: Perl, Tcl, and Python. There are a number of articles comparing these three languages to each other, as well as to compiled languages such a C, C++, and Java (see http://www.python.org/doc/Comparisons.html for a list of articles). After some experimentation with these different languages, we learned that all interpreted languages are not created equal — each has its specific strengths and weaknesses. Of course, they all provide a scriptable framework that is interactive, flexible, extensible, and embeddable but there are style and philosophy differences that make one more compelling than another for a given task.

Perl has the largest user base and is excellent for surprisingly short scripts that do a lot of work, which unfortunately also can be quite challenging to understand. This language offers good support for common application-oriented tasks, whereas Python's elegant, and not overly cryptic syntax emphasizes support for common programming methodology and promotes code readability and thus maintainability. Tcl, like Python, can be used as an extension language and a stand-alone programming language but its support for data structures is rather weak (traditionally everything is a string). Moreover, the lack of modular name spaces before version 8.0 hindered the development of large programs. All these languages span multiple platforms and often provide more platform independence than Java. They all can be extended in C or C++.

We settled on Python for a number of reasons: its concise and almost pseudocode-like syntax; its modularity; its object oriented design; its profiling, debugging, reflection, introspection, and self documentation capabilities; and the availability of a numeric extension allowing the efficient storage and manipulation of large amounts of numerical data. Python is as good a glue as any other interpreted language, and it can be used to develop substantial extension components.

Python

Python is an interpreted, interactive, object-oriented programming language. It provides high-level data structures such as list and associative arrays (called dictionaries), dynamic typing and dynamic binding, modules, classes, exceptions, automatic memory management, etc. It has a remarkably simple and elegant syntax and yet is a powerful and general purpose programming language. It was designed in 1990 by Guido van Rossum. Like many other scripting languages, it is free, even for commercial purposes, and it can be run on practically any modern computer. A Python program is compiled automatically by the interpreter into platform-independent bytecode that is then interpreted. We are running unmodified components written in Python under Linux, Windows NT, 98, 95, IRIX, SunOS, and OSF.

Python is modular by nature. The kernel is very small and can be extended by importing extension modules. The Python distribution includes a diverse library of standard extensions (written in Python, C, or C++) for operations ranging from string manipulations and Perl-like regular expressions, to Graphical User Interface (GUI) generators and including web-related utilities, operating system services, debugging and profiling tools, etc. New extension modules can be created to extend the language with new or legacy code. We describe these extension capabilities below. There are a substantial number of extension modules that have been developed and distributed by the Python user community. These extension modules, sometimes referred to as "packages" or components, include GADFLY, an SQL database manager written in

Python; PIL, the Python imaging library; FNORB and OmniBorker, CORBA compliant Object Request Brokers (ORB) written in Python; Gendoc, an automated documentation tool; and Numeric Python, just to name a few.

The best resource for Python, along with the books that are available, is probably the Python web site (http://www.python.org). It provides access to code, documentation, packages, articles, mailing lists etc. It is also worth mentioning the recent creation of the biopython.org web site, a collaborative software effort for computational biology and chemistry very much like bioperl.

Finally, besides the C implementation of the Python interpreter, there is also a 100% pure Java implementation called Jpython. JPython allows Python use as an interpreted language for programming in the Java world. This interpreter allows initiation of a Java class, and Java code can call Python code. The native extensions first need to be made available in the Java world before they become available in JPython.

The Python Numeric Extension

Numeric Python is an extension module for efficient storage and data-parallel manipulation of numerical data. Using this module, many simple operations which would be too slow in Python can be performed very efficiently (basically as fast as in C) without having to implement the code in C. This module allows us to write extensions in Python (using Numeric) that are efficient enough that they do not need to be re-coded in C or C++.

For example, if the coordinates of the N atoms of a molecule are stored in a matrix of Nx3 floating point values called "Coordinates," operations like the ones demonstrated in the following Python code become syntactically trivial and as efficient as if they were programmed in C or C++:

- Translate the molecule using a 3-vector T. This illustrates data-parallel operations, the 3-vector is added to each 3 vector in Coordinates.

```
>>> Coordinates = Coordinates + T
```

- Compute the center of gravity of the molecule. First sum all values while collapsing the array's first dimension, then divide by the number of atoms.

```
>>> import Numeric
>>> g = Numeric.sum(Coordinates/
len(Coordinates)
```

- Compute the distance from all atoms to a given 3D point P.

```
>>> d = Coordinates - P
>>> # now sum over the 2nd
dimension and take the sqrt
>>> d = Numeric.sqrt(Numeric.
sum(d*d, 1))
```

Performance Issues

Very often, there is a concern about performance when using an interpreted language. We had misconceptions about our performance requirements. Python code runs with reasonable performance sufficient for many common tasks. Having the Numeric extension helps preserving good performance even when working with large arrays of numbers. For example, we developed our first PDB parser for Python as a C extension because we thought Python code would not be efficient enough. The C extension ended up being very complex and became difficult to maintain and extend. So we developed a Python version that comes very close in performance to the C version for parsing a PDB file (a tree-like structure) and computing the connectivity. It was much smaller, simpler, easier to maintain, and platform independent. We feel that the small performance loss was well worth what we gained in flexibility and portability. Since then, we prototype all new extensions in Python and then profile them before deciding what parts really need to be re-implemented in C or C++.

Python as an Integration Tool

Key features of all modern interpreted languages are their extensibility. A Python-based scripting framework can integrate legacy code and be extended with new functionality. There are basically three ways to add new functionality to this scripting framework: implementing it in Python, "wrapping" existing code, or by creating an interface to existing code that has some communication capabilities.

Implementing the functionality in Python

This is the method of choice for all our code development. Indeed, for code that has not yet been written, the high-level nature of Python and the already available extensions make it generally much easier to implement in Python than in C or C++. Even when some code already exists, there are a number of cases where we decide to re-implement in Python. This happens when re-coding in Python requires a relatively small amount of code and the performance of the Python code is expected to be acceptable. This approach provides the major advantage of platform independence.

Wrapping existing C, C++, and Fortran code

There are many cases where it does not make sense to re-implement some legacy code in Python, yet it is desirable to have access to this code from Python. This requires "wrapping" the legacy code for Python. Let us consider the following simple example: we have a function "foo" that takes an integer as an argument and returns a float. In the Python world integers and floats are objects. To call the function **foo** from Python, we have to write a function in C that will take a Python object as an argument; extract the integer value from this object; call the function **foo** for this integer value; package the returned float into a Python object representing a float; and finally, return that object to the interpreter. This code is called the "wrapper code." Once we have the wrapper code, we can compile it along with the function **foo** and create a shared object (dll in the Windows world) which can then be imported into a Python interpreter. Of course, this extension is now platform dependent. This process of wrapping code can be automated somewhat using SWIG, Simple Wrapper Interface Generator (http://www.swig.org). SWIG can generate wrapper code for several interpreted languages like TCL, Perl, Guile, and, of course, Python. Recently, support for Java has been added, too.

This approach is very useful to

extend Python with Application Program Interfaces (API) of existing software. Once such an API has been wrapped, it can be made available to the community: the number of such extensions is large and growing. (Oracle, mySQL, OpenGL, DCOM, etc.).

Interfacing Code

For legacy code that was designed with some communication capabilities, such as through sockets or using a standard communication protocol (HTTP, NNTP, SMTP, etc.), the support provided by the standard Python modules for these protocols makes is generally quite easy to write a client in Python.

Finally, a Python interpreter can be embedded in an application as an extension language. We have described such an approach for AVS (2). Besides adding scripting capabilities to the program in which the Python interpreter is embedded, this also makes all the tools ported to Python immediately available within this program.

Python Extensions We Have Developed

Over the last couple of years, we have developed a number of Python modules or components to deal with many aspects of our daily work. Using SWIG, we have wrapped codes dealing with tasks such as molecular surface computation, molecular mechanics and dynamics, electrostatic calculations, protein-protein docking, protein-ligand docking, etc. Besides these "traditional" computational methods, we have also wrapped a number of less conventional tools for molecular modelling such as a convex hull calculation tool, a rapid collision detection of polygonal models library, several mesh decimation algorithms, a general extrusion library, etc. These modules are all platform dependent: we are mainly using them on SGI and Sun computers. We have also developed two platform independent packages in Python: DejaVu/DejaVu Framework which is a general 3D geometry visualization package, and BioChem which provides support for manipulating and visualizing molecules.

DejaVu is a package written in Python for the visualization of 3D geometry using the OpenGL library. It provides a set of classes describing objects such as viewer, cameras, lights, clipping planes, color editor, trackball, geometries, etc. The Viewer class is a fully functional visualization application providing control over a number of rendering parameters such as depth cueing, global anti-aliasing, rendering modes (points, lines, polygons), shading modes (flat, gouraud), multiple light sources, arbitrary clipping planes, etc. An instance of a Viewer maintains a hierarchy of objects in which rendering attributes can be inherited. DejaVu also provides a number of standard geometries including lines, indexed-lines, indexed-polygons, triangle and quad strips, spheres, cylinders, and labels. This list can be extended by sub-classing the geometry base class. DejaVu makes it very trivial to add visualization capabilities to any object developed in Python.

DejaVuFramework is a "sub-package" of DejaVu that provides support for building visualization applications in which the overhead of creating new commands and their associated GUI is minimal. This framework provides support for loading dynamically commands from libraries as they are needed. It has been designed to be specialized and extended in a modular way.

The BioChem package provides classes to read molecules in a number of file formats and build a tree-like hierarchical structure reproducing the molecule's structure. For instance, a protein molecule is represented using the following four levels: molecule, chain, residue, and atom. The number of levels can be modified. For instance, we have added a level between chains and residues to represent the secondary structure of the protein. All nodes in that tree-structure can be extended dynamically: i.e., to add a property like a radius to an instance of an atom called "A," one simply assigns the value to a member data of the atom: A.radius = 1.7.

This structure allows dynamic extension of any node and performs all kinds of selections. It has proven to be very powerful and provides a high-level interface to the molecule. The following code initiates a Protein object and builds its representation from a PDB file. Several selections and manipulations illustrate some the basic capabilities of these classes:

```
>>> from BioChem.protein import
Protein
>>> from BioChem.pdbParser import
PdbParser
>>> protein = Protein()
### create a protein object
>>> ### read the file with a giv-
en parser object
>>> protein.read('1crn.pdb', Pdb-
Parser() )
>>>
>>> ### Once the protein is
loaded, we can operate on that
>>> ### structure
>>> ### TreeNodeSets behave like
Python arrays
>>> residues8Trhough15 =
protein.chains[0].residues[8:16]
>>> allAtoms =
protein.chains.residues.atoms
>>> print residues8Trhough15
<ResidueSet instance> holding 8
Residue
>>>
>>> ### select atoms all atoms
with their name in list:
>>> bbnames = ['N', 'CA', 'C',
'O']
>>> backBone = allAtoms.get(lamb-
da x, names=bbnames: x.name in
names)
>>>
>>> ### perform boolean
operations over set:
>>> sideChains = allAtoms - back-
Bone
>>>
>>> ### add new members on the
fly to any object or set of >>>
### objects: here we add the a
member called "bb"
>>> ### to all atoms and set it
to 1
>>> ### for backbone atoms and 0
for side chain atoms
>>> allAtoms.bb = 0
>>> backBone.bb = 1
>>>
>>> ### build bonds using atomic
distances
>>>
protein.buildBondsByDistance()
```

A Molecule Viewing Application

Building on top of the BioChem and DejaVu packages, we have developed a molecular viewer (Figure 1). This viewer has most of the features usually expected in a molecule viewer: stick and cpk representation; different coloring schemes (by atom, by residue type, by chain, by molecule, by properties); measuring tools; atom identification by picking; support for multiple molecules; secondary structure representation; and user definable sets of atoms, residues, chains, and mole-

cules. In addition to these traditional features, it is dynamically extensible, i.e., new commands can be developed independently and placed in libraries. The Viewer inherits from the DejaVu Framework the capability to dynamically import these commands as needed. In fact, all commands in that viewer have been developed based on this principle. This provides a way to add features to the application that is incremental and well suited for team development. This approach avoids the "feature overload" problem, i.e., overloaded menus cluttered with commands that are irrelevant for the problem at hand. Customization files allow users to specify which commands should be loaded when the application starts, among other things. This allows the user to define a number of molecular viewing applications just by creating different customization files, each loading different sets of commands. We have written a surface calculation command and one to selectively display pieces of a surface corresponding to a subset of atoms. In this viewer, all geometries relate to the underlying molecules, therefore, a subset of atoms can be used to: partially display/undisplay; selectively label; locally color using different schemes; and any of the geometries. And vice-versa: picking done on any geometry (molecule, surface, secondary structure) is mapped back onto the molecule. This application also provides a Python Shell. This Python-aware, type in widget is the standard input and standard output of the python interpreter running the application. From the Python Shell, any command can be called interactively and scripts operating on the application can be written. Commands log themselves into a file as they are executed. This file can be saved, edited, and played back to restore a previous session. This application runs unmodified on any platform that has a Python interpreter with the following extensions: Numeric, Tkinter/Togl, and OpenGL.

Maybe the most interesting feature of this application is that creating the application actually comes down to writing the commands, and writing commands is made easy by the built-in support provided by the DejaVu Framework.

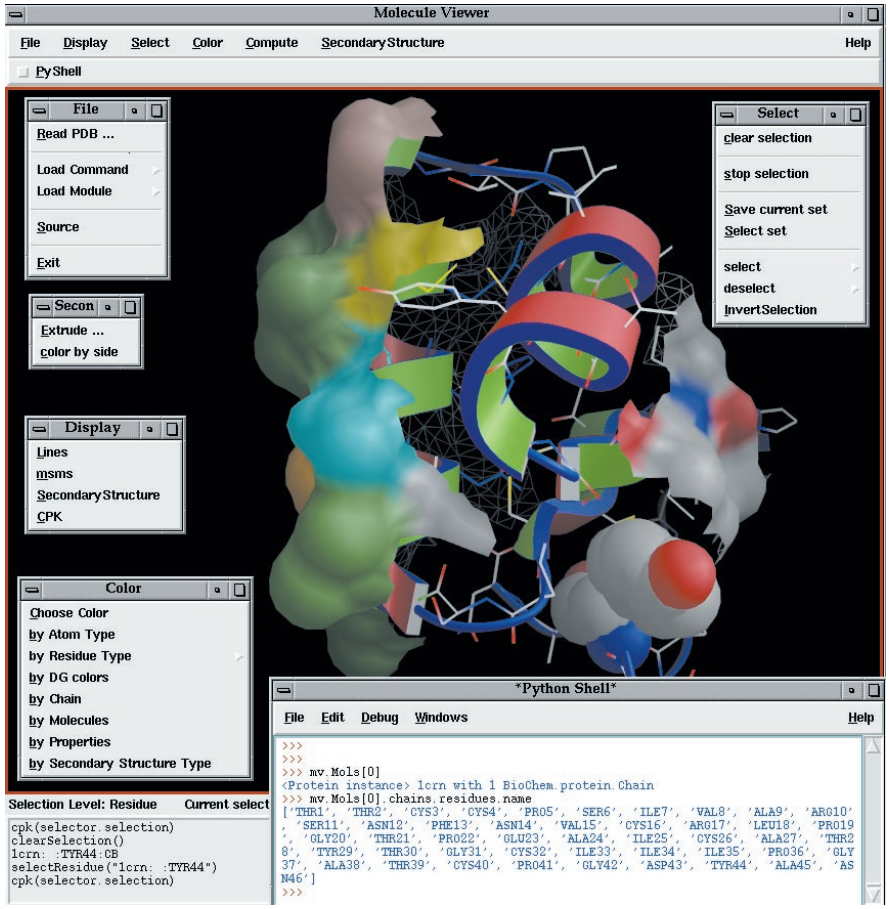

Figure 1. The molecule viewer application showing the protein Carmbin (1 crn) with its secondary structure shown as a ribbon. The molecular surfaces corresponding to helix1 and sheet2 are displayed and colored; using, respectively, the "RASMOL residue" coloring scheme and the "by atom type" coloring scheme. Some pull-down menus have been torn off to show the commands they provide. This set of commands can be extended dynamically by loading modules and commands from libraries.

Conclusion

We are convinced that this "language-centric" or scripting framework approach has a lot of strengths and benefits. We already witnessed a dramatic increase in our productivity, as well as a high level of code reuse. Of course, this approach is not specific to Python and could be reproduced with any interpreted language and even with compiled languages. However, the fact remains, that it did not happen with any of the other languages we have used previously. The most important change for us has been to shift from writing programs or scripts to writing modules or components. This concept is well known to the community of component-based software development; but in a molecular modelling research environment, with limited resources for the design and implementation of software tools, Python has been instrumental. It has placed these advanced software development techniques within our reach.

References

1. Michel F. Sanner et al. (1998). Integrating Computation and Visualization for Biomolecular Analysis: An example using Python and AVS. Proc. Pacific Symposium in Biocomputing '99. p. 401-412.
2. J.S. Poulin. Reuse: been there, done that. Technical Opinion. Communication of the ACM. May 1999, 45:98-100.

Quantum Chemistry Program Exchange

Donald B. Boyd, Ph.D., Department of Chemistry, Indiana University-Purdue University at Indianapolis (IUPUI), 402 North Blackford Street, Indianapolis, IN 46202-3274, boyd@chem.iupui.edu

QCPE: End of an Era or a New Chapter?

At the end of July 1999, Mr. Richard W. Counts retired from the Quantum Chemistry Program Exchange (QCPE), which he had managed for the last 30 years. Not many months earlier, his long time coworker, Dr. Margaret (Peggy) Edwards, also retired from QCPE. Options for the future of QCPE are presently being considered at Indiana University at Bloomington (IUB). QCPE has a long and distinguished history in the dissemination of software used by computational chemists. Many younger computational chemists may not fully appreciate the role QCPE played in the development of the field, so it is worthwhile here to briefly review QCPE's purpose and evolution.

QCPE was founded at the inspiration of Professor Harrison Shull, a quantum chemist at IUB. His vision was to have a central, international repository of software used by quantum chemists. At meetings and elsewhere, he convinced his fellow theoreticians of the advantage of exchanging computer programs. He pointed out that it was wasteful of the time of graduate students at every university to have to write a program to do the same quantum mechanical integral calculation as had already been programmed elsewhere. A second motivation for setting up a library of shared software was to create a relatively permanent repository. If a graduate student finished a thesis and left a university or if a professor changed research interests, software they had written would not be lost or lie unused on some forgotten shelf. A third motivation for a central repository was to create an intermediary between the code writers/owners and users. Quantum chemistry professors whose students had created useful programs often shared copies with other research groups. However, the users in the other groups might not understand the requirements of operation or the limitations for getting useful results. Hence these users would constantly be asking the developers for help. For widely used programs, such requests for help could consume a significant amount of time and distract the developer from other work. So, someone at a central depository who had a basic knowledge of quantum chemistry could field some of these questions from the users.

There was general agreement that an organization like QCPE would be useful to the community of theoreticians, but it took the hard work and dedication of Dr. Shull and his colleagues at IUB to bring QCPE to reality. QCPE was launched in April 1962 with 23 pieces of software, mostly quantum chemistry subroutines, ready for distribution. The chemistry department at IUB provided the infrastructure for QCPE to operate. Among the individuals who helped in the early days of operation were Dr. Keith Howell, Shull's postdoctoral associate from England, and later, Dr. Franklin Prosser. (Frank Prosser, who has a Ph.D. in physical chemistry, retired from the computer science department at IUB in 1999.) Dr. Stanley A. Hagstrom, another theoretician at IUB who is now emeritus professor, also played a vital role in the birth of QCPE, especially at the technical level. He developed initial submission procedures, documentation guidelines, and distribution procedures. In late 1964, a significant event occurred in the life of QCPE when funding was secured from the Directorate of Chemical Sciences, Air Force Office of Scientific Research, U.S. Office of Aerospace Research.

QCPE served as a conduit through which individual researchers could donate their programs. The programs were checked to make sure that they compiled, performed as claimed, and contained at least a minimal amount of documentation in the form of "comment cards" or write-ups. Then the availability of the programs was announced through QCPE's catalog, and the software was sent to individuals who paid the modest distribution and handling costs. In the 1960s, the software was distributed on computer cards and magnetic tape. Generally the programs were written in whatever was the current version of FORTRAN, and they ran on mainframe computers of International Business Machines (IBM), Control Data Corporation (CDC), and various hardware companies.

In the 1960s, QCPE regularly published a list of its members, which was several pages long. QCPE also published a quarterly newsletter with news, announcements of new members, and progress reports from individual theoretical chemistry research groups around the world. By 1965, QCPE membership had grown to 425 individuals worldwide, it had a library of 71 programs, and 500 copies had been distributed. In that period, the most frequently requested programs were for running extended Hückel molecular orbital calculations and for evaluating two-electron integrals for ab initio calculations. In 1967, Dick Counts, with a physics background and a Master's degree, was hired from IUB's Aerospace Research Applications Center to become administrator of QCPE.

All through the 1970s, QCPE continued growing and provided exemplary service to the community of theoretical chemists with an ever expanding library of programs. Some of the deposited programs ran without problem, but others were written very specifically for one machine or just one machine configuration. Dr. Hagstrom and other colleagues at IUB provided assistance to QCPE by getting such programs operational on other machines. We do not have space here to list all the programs in the library, but among the ones then popular was CNDO/INDO program (QCPE 141) from Paul Dobosh in John Pople's group at Carnegie-Mellon University. Also, Pople's group released Gaussian 76 to QCPE in 1978 (QCPE 368). (For a complete list of QCPE software,

see *http://qcpe.chem.indiana.edu/*.) In 1971, QCPE received a grant from the National Science Foundation, which put the organization on a solid financial footing. Starting in 1973, QCPE became self-supporting. A very modest annual membership fee was charged to members. Users purchasing software at the modest distribution cost was the other main source of revenue.

In 1981, the *QCPE Newsletter* was formalized as the *QCPE Bulletin*. It was published quarterly and included short citable articles, editorials, announcements of newly deposited software, and news of interest to the community. Richard Counts was editor and Peggy Edwards, a former secretary at Eli Lilly and Company with a Ph.D. in English, was assistant editor. A QCPE Advisory Board was appointed to help offer different perspectives on the interests and needs of the membership. Also in 1981 when Prof. Shull moved to Monterey, CA, Prof. Hagstrom assumed the role of Director of QCPE, while Mr. Counts and Dr. Edwards continued to manage the day-to-day operations. From April 1980 to April 1981, 451 programs were distributed to the United States, 212 to West Germany, 138 to Great Britain, 106 to Japan, and 77 to Switzerland. Also in 1981, the short-lived U.S. National Resource for Computational Chemistry (Lawrence Berkeley Laboratory, Berkeley, CA) ceased operations and turned its software collection over to QCPE.

Besides serving as a repository of software and producing the *Bulletin*, QCPE performed another valuable service in the 1980s. Mr. Counts organized annual summer workshops on Practical Applications of Quantum Chemistry. Most of these intense week-long courses were held at IUB, but one was held in Oxford, England and another in Marlboro, MA. The workshops were taught by practicing computational chemists and included hands-on experience running important programs in QCPE's holdings. Back in the early 1980s, input data was still prepared on IBM punch cards, and the jobs were run on the mainframes at IUB. The workshops exposed 20 to 25 individuals each year to computational chemistry tools. Not all the individuals taking the courses were newcomers to the field; many were experienced users who came to learn about the latest programs and the advantages and pitfalls of each method. The QCPE workshops were so effective at training users that other universities and organizations emulated them.

Many popular programs, such as the molecular mechanics program MM2 from Allinger's group and the semiempirical molecular orbital programs (MINDO and MNDO), appeared in QCPE's catalog. However, a significant milestone occurred in May 1983 when the MOPAC program (a General Molecular Orbital Package) was deposited by Dr. James J.P. Stewart. He was on extended leave from the University of Strathclyde, Scotland, and working as a postdoctoral associate in M.J.S. Dewar's group in Austin, TX. MOPAC (QCPE 455) became by far the most popular and influential program in QCPE's offerings. The appearance of MOPAC coincided with the manufacture of the hugely successful VAX 11/780 superminicomputers from Digital Equipment Corporation (DEC). The late 1980s thus saw an increasing number of QCPE holdings that ran on departmental computers. Many programs that were originally developed for large mainframes were ported to these less expensive machines and eventually to personal computers.

Another step in QCPE's history occurred in 1984 when Professor Ernest R. Davidson was invited to move his group to IUB from the University of Washington, Seattle. Ernie Davidson was named to the Advisory Board and replaced Stan Hagstrom as faculty advisor. Again Mr. Counts and Dr. Edwards continued to manage the day-to-day operations. In its heyday, QCPE distributed about 2,500 programs per year. Mr. Counts hired students to help him and Dr. Edwards with the heavy workload. The software catalog became so thick that it was broken into subcategories. A standardized format for citing QCPE software was published in the *QCPE Bulletin*, and indeed QCPE programs have been widely cited in the scientific literature. The name of the organization and the bulletin was shortened from Quantum Chemistry Program Exchange to simply QCPE is due to the fact that the software library had evolved from being just about quantum chemistry to molecular modelling in general. Organizationally, QCPE was under the purview of the IUB Chemistry Department, but Mr. Counts ran the operation essentially independently. However, QCPE was primarily a service to the community. It was not a big revenue generator for the Chemistry Department.

One of the reasons for the popularity of QCPE was that most of the programs distributed were in the form of source code. By obtaining source code, other researchers could extend, modify, and perhaps improve a piece of software. In contrast, most software companies, which sprang into existence in the 1980s to serve the growing computational chemistry market, rarely distributed their source code. Nevertheless, a number of factors undermined the important role QCPE was playing. The 1980s and 1990s witnessed the commercialization of software by relatively large companies in the computational chemistry business. Customers had to buy commercial versions of MOPAC, AMPAC, MM3, Gaussian, and other popular programs in order to obtain the latest versions with the most features and most bugs fixed. The QCPE software holdings became less relevant to the present-day mode of computing with graphical user interfaces. By 1990, the QCPE library did have some programs with GUIs. The library also had some elaborate molecular modelling programs qualifying for the name "system" or "package." Another major trend in the 1990s was the emergence of the Internet which permitted individuals independent ways of distributing software they produced. The flow of new programs being deposited in QCPE gradually diminished. The number programs being requested also slowed in the 1990s, although exact figures are unavailable. Software had been deposited by American chemists as well as researchers in many countries besides the U.S. However, distribution of programs in the last 10 or 15 years was largely *out* of the U.S. The Japanese remained some of the best customers

of QCPE's holdings. Another trend impacting QCPE was the fact that users wanted and expected technically supported software, i.e., they wanted to be able to call up a toll-free telephone number and ask questions about the operations of a program. Mr. Counts provided such support on an ad hoc basis.

In 1989, QCPE became reachable via e-mail over BITNET. After about 1991, the QCPE Advisory Board did not meet and was eventually dropped. Starting in 1993, QCPE made its catalog available by file transfer protocol (ftp) and began distributing software that way. Also in 1993, QCPE acquired an e-mail address on the Internet. Whereas at one time 2000 members were receiving the *QCPE Bulletin*, in the last few years the membership has slipped toward 1,000.

Presently, the exact role of QCPE at IUB is still being sorted out. Dr. Marty Pagel, Director of Information Technology in the IUB Chemistry Department, has been designated to handle QCPE's operations in addition to his other duties. Some software continues to be deposited at QCPE. But the continued viability of QCPE is unclear. The QCPE software library is presently approaching 775 programs for mainframes and workstations, plus about 125 additional programs for desktop computing. This collection represents hundreds of thousands of lines of source code, much hard work, and creativity. These programs continue to be distributed; presently, QCPE receives about 15 orders per month. Publication of the *QCPE Bulletin* has been suspended, but it may be made available through an expanded QCPE home page. Some other exciting developments are planned as QCPE becomes more web-based in the year 2000.

When Mr. Counts retired in July, he wanted no special event in his honor. The last major tribute to QCPE was at a symposium organized "in honor of R.W. Counts for service to the field of computational chemistry" held by the Computers in Chemistry Division (COMP) of the American Chemical Society at the 207th National Meeting, March 13-17, 1994, San Diego, CA.

All the people who were involved in organizing and running QCPE, plus all the individuals who deposited programs, participated in the workshops, did behind-the-scenes work to get programs tested and usable, or otherwise served the organization, deserve profound thanks. They and QCPE helped bring about the birth of computational chemistry as it is known today.

NEWS BRIEFS

Research Collaboratory for Structural Biology Considers NMR

The Research Collaboratory for Structural Biology (RSCB, *http://www.rcsb.org/pdb/*) formed a task force to provide it with advice on aspects of the PDB that are unique to the NMR. Among these issues are protein nomenclature, multiple ID codes, ensemble data, constraint files, and validation concerns. Task Force members are:

- R. Andrew Byrd – National Cancer Institute - FCRDC
- Iain Campbell – Univ. of Oxford
- Marius Clore – NIH-NIDDK
- Peter Domaille - Dupont Pharmaceuticals
- Juli Feigon – Univ. of California, Los Angeles
- Robert Kaptein – Univ. of Utrecht
- John Markley – Univ. of Wisconsin
- Michael Nilges – EMBL, Heidelberg
- Michael Summers – Univ. of Maryland, Baltimore
- Peter Wright – Scripps Institute
- Kurt Wüthrich – Swiss Federal Institute of Technology

The Task Force held a short organizational meeting at the Keystone Frontiers of NMR in Molecular Biology VI Symposium in Breckenridge, CO, in January, followed by a one-day workshop in Rockville, MD, on April 23, 1999. The workshop included discussions on the deposition tool and validation needs.

The BioMagResBank (BMRB, *http://www.bmrb.wisc.edu/*) has been receiving a substantial portion of their chemical shift data through PDB/EBI, and the RCSB will continue to accept chemical shift data and send it on to BMRB. The BMRB and the RCSB are committed to working together to develop and implement a single interface for NMR deposition of coordinates and experimental data.

Here are some of the highlights of the discussions:

Proton Nomenclature

Task Force members were in consensus as to the advisability of adopting IUPAC nomenclature (1,2) in the coordinate files, even though this would involve substantial changes in the files because IUPAC nomenclature will require:

- Changing the numbering of methylene hydrogens.
- Establishing aromatic ring numbering on a model by model basis since CD1 and CE1 designations are dependent on c2 torsion angles; further discussion may be warranted with respect to model by model numbering.
- Correction of side chain-NH_2 hydrogen nomenclature in Asn and Gln residues (currently done in the RCSB validation procedure).
- Correction of Arg side chain nitrogen nomenclature as needed (currently done in the RCSB validation procedure).

These changes will establish consistency with the BMRB nomenclature, but will create inconsistencies with existing literature and constraint files. Efforts will have to be made to document these differences.

Multiple ID Codes

There was consensus for one accession code for a single study, i.e., put all members of the ensemble and the minimized average structure under one code rather than the multiple codes that now exist. In the old PDB files, the atoms have been numbered sequentially from model to model, with the result that the field limit for the atom number often necessitated the creation of multiple files with multiple accession codes. In the newly deposited PDB files, the atom serial numbers are sequential within each model of the ensemble so that all models of the ensemble have the same atom serial numbers. The CONECT records,

provided for the first model, are now applicable to all the models.

The minimized average structure will need to be identified with a suffix tag so that those structures are not inadvertently downloaded and treated as just another member of the ensemble. A single accession code might include the ensemble and the average, or representative structures.

Ensemble Issues

The following ensemble issues were raised, but no consensus was reached:

- Should there be a limit placed on the number of models submitted or a suggested limit?
- Should the H-bonds, salt bridges, helices, sheets, and turns be defined for the ensemble as a whole or for each model? Currently these parameters are defined for the first model. If defined for the whole ensemble, how should a consensus value be derived?
- Will IUPAC aromatic ring nomenclature present a problem if nomenclature changes from model to model?

Validation Issues

The Task Force reviewed the current RCSB validation process for the NMR data, which relies on Procheck-NMR3 for protein ensembles, Procheck4 for the minimized average protein structures, and Nucheck for nucleic acid structures (ADIT Validation Server: *http://pdb.rutgers.edu/validate/*). The Task Force felt that these programs did a good job of alerting authors to geometric anomalies that might signal a structural problem.

There was some interest in constructing a template for deposition of the chemical shift data and NOE peak tables including peak areas or volumes and a listing of atoms that float during structure calculations. The adoption of a standard common format would not be possible since the program contents are different, e.g., some programs use pseudo atoms and others do not, the types of constraints vary, etc. Validation of experimental data will require further consideration.

Data Deposition

The ADIT deposition and annotation system is built on the mmCIF dictionary and the NMR STAR extensions. It provides precise definitions and examples and defines data relationships, data type, range restrictions, allowed values, etc. The dictionary is organized in table-like structures called "categories."

References

1. J.L. Markley, et al. 1998. Recommendations for the Presentation of NMR Structures of Proteins and Nucleic Acids, Pure & Appl. Chem. 70:117-142.
2. C. Liebecq. 1992. Compendium of Biochemical Nomenclature and Related Documents, 2nd ed., Portland Press: London and Chapel Hill, NC.
3. R. A. Laskowski et al. 1996. AQUA and PROCHECK-NMR: Programs for Checking the Quality of Protein Structures Solved by NMR. J. of Biomol. NMR 8:477-486.
4. R.A. Laskowski et al. 1993. PROCHECK: A Program To Check the Stereochemical Quality of Protein Structures. J. Appl. Cryst. 26:283-291.

The EMBL-European Bioinformatics Institute, Macromolecular Structure Database and the PDB

Geoffrey J. Barton, European Molecular Biology Laboratory Outstation, European Bioinformatics Institute,- Wellcome Trust Genome Campus, Hinxton, Cambridge, CB10 1SD, UK, geoff@ebi.ac.uk

The European Bioinformatics Institute (EBI) (http://www.ebi.ac.uk) an outstation of the European Molecular Biology Laboratory (http://www.embl-heidelberg.de) is a center for research and services in bioinformatics. The EBI manages the EMBL nucleotide sequence database (a collaboration with NCBI in the U.S. and DDBJ in Japan), SWISS-PROT/TrEMBL protein sequence database (a collaboration with the Swiss Institute for Bioinformatics), and the EBI-MSD Macromolecular Structure Database, a collaboration with the RCSB-PDB in the U.S. as well as several other smaller databases. The EBI-MSD was established at the end of 1996 and provides services for the deposition, management, and search of data on macromolecular structure in Europe. This article gives a brief overview of current services available from EBI-MSD, and a hint of what is to come during late 1999, 2000 and beyond. For up to date information on the EBI-MSD, including job opportunities with the project, see *http://msd.ebi.ac.uk.*

Deposition of New Structures to the PDB at EBI

EBI-MSD are developing a new system (EC-Dep) for the deposition of macromolecular structures to the PDB called. This will be phased in from autumn 1999. In the meantime, you may deposit structures to the common RCSB/EBI-MSD PDB archive using AutoDep at EBI (*http://autodep.ebi.ac.uk*). If you need help completing the forms, please e-mail pdbhelp@ebi.ac.uk. EBI-MSD staff will normally reply quickly during 09:00-17:00 GMT working hours. A PDB ID code will be issued by e-mail as soon as the AutoDep session is completed. All depositions made at EBI since 15 June 1999 have been processed by EBI-MSD staff. Entries that have been reviewed by the authors and reached their release date are forwarded each Monday to the RCSB for general distribution through the RCSB ftp site on Wednesday. You can also deposit to the PDB at the RCSB in the U.S. using the ADIT system (http://pdb.rutgers.edu). Data submitted to the U.S. site are processed by the RCSB in the U.S. It does not make any difference where you deposit your data since EBI-MSD and RCSB work closely to maintain consistent data processing procedures. You should choose the deposition site that is most convenient for your time zone.

Searching the PDB

There are a very large number of services on the WWW that provide access to the PDB and data derived from it. In this section, we only mention tools that are directly related to the EBI-MSD, the former PDB at BNL, and the RCSB.

3DB Browser

EBI-MSD are continuing to support querying of the PDB by the tools: see *http://pdb-browsers.ebi.ac.uk/pdb-bin/pdbmain* and *http://pdb-browsers.ebi.ac.uk/pdb-bin/pdblite.* However, we are not enhancing or developing them.

New EBI-MSD Database and Tools

Extensive work at EBI on clean-up of the legacy PDB files has led to the development of the following services:

• PQS:
http://msd.ebi.ac.uk/Services/Quaternary/quaternary.html
Standard PDB files only contain the contents of the asymmetric unit (ASU). This may or may not include the complete biologically active molecule. For example, the ASU may only contain a monomer, but the true state is a dimer that is only revealed by applying crystallographic symmetry. PQS overcomes this problem by providing access to the probable quaternary structures of the macromolecules.

• 3Dseq
http://msd.ebi.ac.uk/Services/Sequence/sequence.html provides a direct mapping between the PDB and SWISS-PROT sequence database at the residue level. Residue differences between SWISS-PROT and PDB are annotated.

• Unpublished References
http://msd.ebi.ac.uk/Services/UnPubRef/unpubref.html provides links to references for structures listed as Unpublished.

These three services are being incorporated into a new Oracle relational database system under development at EBI-MSD. New search tools based on the relational system will be announced as they become available.

PDB FTP Site at EBI

The RCSB maintain the ftp site that contains PDB format data files. At EBI we provide a full mirror (*ftp://ftp.ebi.ac.uk/pub/databases/pdb*) of this site in the same form as it was maintained by the former PDB at BNL. The EBI-MSD will also support a new simplified RCSB ftp site structure.

Future Services

The cleanup of PDB performed at EBI will be merged with complementary efforts at RCSB to form a single consistent archive of data on macromolecular structure. The cleanup standardizes naming as well as placing all coordinates in a consistent frame. EBI-MSD plans to offer the structural database in a relational form with a programmable interface for local use as well as through a web interface at EBI. The new system will make it much easier to ask complex questions across the entire PDB archive than is possible with current tools.

In addition to the work on the existing atomic resolution data derived from X-ray and NMR techniques, EBI-MSD are working with the single particle Electron Microscopy community to merge the existing database of low resolution structures from EM with the atomic resolution data. When X-ray, NMR, and EM are coupled together in a single database system, they will provide a more complete picture of macromolecular structure and function.

Columbus Molecular Software, Inc. Releases LeadScope™ 1.0

Columbus Molecular Software, Inc. (*www.columbus-molecular.com*) has released LeadScope™, their new decision support tool for medicinal chemists. This commercial release incorporates insights gained from a collaboration with Pfizer.

Unlike general-purpose decision support tools, LeadScope has been developed specifically to support a variety of tasks performed by the medicinal chemist doing drug discovery. The software incorporates a knowledge base of structural features, a chemistry-aware analysis engine, and a highly visual and interactive interface to help explore the processed structures and their associated properties. This eliminates the need for medical researchers to customize existing analytical tools to process and extract this information, freeing more time for actually performing drug discovery. Because of LeadScope's ability to provide insight into structural and property data, users may have an interest in reviewing old data.

Columbus Molecular Software, Inc. is an informatics company focused on the research, development, marketing, and support of chemically intelligent decision support systems.

Biomer: A Java Molecular Modelling Package

Neill White, The Scripps Research Institute, Department of Molecular Biology, 10550 North Torrey Pines Road, La Jolla, CA 92037, nwhite@scripps.edu

Biomer is a new on-line molecular modelling package written in Java. Motivated by a dearth of free, GUI-based model builders, Biomer was originally designed as a means to easily generate initial PDB structures of nucleic acids and polynucleotides. Later, the rudimentary model building features were enhanced and augmented with gradient descent and simulated annealing energy minimization techniques.

The model building facilities of Biomer allow for a detailed description of the geometric secondary structure parameters of polynucleotides. The user can completely specify all the rotational and translational helical parameters as described in reference 1, as well as the sugar pucker of the furanose ring. In addition, the polynucleotide builder can build both single- and double-stranded regions of DNA and RNA and offers default values for a number of well-defined nucleic acid forms. The user also has the ability to add unusual base pairs (both minor nucleotides and non-complementary bases) in place of complementary Watson-Crick base pairs.

The polypeptide and polysaccharide builders allow the user to completely specify the relevant backbone torsion angles phi, psi, and omega. The polypeptide builder includes 24 different amino acid options in addition to N- and C-terminating groups. Torsion angles for well-known structural motifs such as the alpha helix and beta sheet are provided, as well as sets of torsion angles describing secondary structure features that span several amino acids (e.g., gamma turns). The polysaccharide builder includes 13 sugar monomers and the ability to specify the geometry of the linkage and the connectivity between successive monomers.

The energy minimization features include gradient descent methods (steepest-descent and conjugate-gradient) and simulated annealing with molecular dynamics using the Verlet algorithm. Several variants of the conjugate-gradient method are available

as options including Polak-Ribiere, Fletcher-Reeves, Hestenes-Stiefel, and Powell's method. The energy function is given by the original all-atom AMBER force field (2).

In addition to canned structure generation and geometry optimization, Biomer also has capabilities as a molecular editor and viewer. Atoms and bonds can be added or deleted and a number of rendering options are available (spacefill, polytubes, ball-and-stick, etc.). A number of geometric measurements can be performed with mouse clicks, such as measure the distance between atoms, and the angles and torsion angles between triples and foursomes of atoms.

Similar programs exist, but few offer the benefits of Java cross-platform compatibility and the ability to run over the Internet without installing on the local hard drive. Typically, one wishes to create a starting structure to simulate in a large package such as AMBER or Charmm. Biomer provides a quick and easy way to generate such structures.

References

1. S. Diekman. 1989. Definitions and Nomenclature of Nucleic Acid Structure Parameters. J. Mol. Biol., 205:787-791.
2. J. Comp. Chem 7(2):230-252,1986. Biomer home page: *http://www.scripps.edu/~nwhite/Biomer*

Chemical Computing Group Announces Port of MOE to Silicon Graphics Visual Workstations for Windows NT

Chemical Computing Group Inc. (*http://www.chemcomp.com*) announced the addition of the SGI 320 and 540 visual workstations for Microsoft Windows NT from SGI to the list of platforms supported by the Molecular Operating Environment (MOE).

Researchers can use advanced digital media and graphics features, including stereoscopic viewing capabilities. MOE has embedded language architecture, built-in out-of-the-box applications, source code for easy adaptation, and a platform for rapid methodology application prototyping. It currently runs on SGI UNIX, DIGITAL Alpha Systems, SUN, Microsoft Windows 95, Windows 98, and Windows NT. MOE's built-in suites of applications cover high throughput screening data analysis, combinatorial chemistry library design, protein and homology modelling, and material science as well as molecular modelling and simulation.

Functional Genomics Consortium Collaboration Between MSI & SGI

Molecular Simulations Inc. (MSI, *www.msi.com*) announced the results of a collaborative project with SGI on functional genomics. Twenty-two genomes were processed, which generated new 3D annotations for approximately 27,000 sequences. Three-dimensional annotations are instrumental in understanding the function of genes and their protein products. New automated technology for high-throughput protein function assignment and annotation from MSI was used to analyze genomic sequences from various public sources and generate functional assignments for their protein products. MSI is delivering this new technology through two new software applications: Gene Atlas and AtlasBase.

Gene Atlas is a high-throughput automated pipeline for functional annotation of protein sequences on the genomic scale. AtlasBase is the database that will store the results of Gene Atlas. Gene Atlas and AtlasBase, as well as the data results of the MSI-SGI collaboration, form the basis of MSI's new Consortium for Functional Genomics.

SGI (*www.sgi.com*) provided a 256-processor Origin 2000 supercomputer on which to run comparative protein analyses on the 22 genomes that generated the 3D protein models. The analysis of around 75,000 protein sequences resulting in prediction and functional annotation of 57,000 validated 3D structural models took a week.

SGI Close to Selling Interest in Cray Unit

SGI is planning to sell most of its Cray supercomputer unit. SGI hopes to reach an agreement with a noncomputer maker that will take a majority stake in the Cray unit.

SGI acquired Cray in early 1996. The market for Cray's vector supercomputers had started to decline even before SGI purchased the company, and it has continued to go down. The part of Cray's original business that has thrived was a unit that made computers based on Sun Microsystems Inc.'s UltraSparc microprocessor, which was purchased by Sun from SGI in 1996.

Meanwhile, the Cray division of SGI has been developing a computer, the Cray SV2, jointly with the National Security Agency that will combine vector and scalar processors. The machine will be designed to handle operations for national defense, but it also could be used in modelling complex systems. The Cray SV2 will be able to handle 30 trillion computations a second, and SGI expects to introduce the new machine by the middle of 2002.

Wavefunction and Schrödinger Release Graphics and Modelling Package for Chemists

Wavefunction (*www.wavefun.com*) and Schrödinger (*www.schrödinger.com*) have released TITAN, combining the speed of Schrödinger's Jaguar software with the flexibility and ease of use of Wavefunction's Spartan software.

TITAN is aimed at both computational chemists familiar with MM and QM, and experimental chemists.

TITAN allows chemists to:

- Construct complex organic, organometallic, and inorganic molecules. Obtain transition states from an extensive reaction database.
- Perform calculations with a wide range of computational methods including molecular mechanics (SYBYL, MMFF), semi-empirical molecular orbital (AM1, PM3 including parameters for transition metals, MNDO, MNDO/d, AM1-SM5.4), Hartree-Fock molecular orbital, density functional (SVWN, BP, BP86, B3LYP), and LMP2. Hartree-Fock, density functional, and LMP2 calculations support both all-electron and pseudo-potential basis sets with extensions for polar-

ization and/or diffuse functions.
- Display multiple model styles. Calculate properties, volumes and surface areas, dipole moments, and atomic charges.
- Present 3D isosurface and 2D slice displays of electron densities, spin densities, electrostatic potentials, and molecular orbitals.
- Map properties onto a surface.
- Analyze results in a spreadsheet.
- Drive coordinates to simulate chemical reactions, search conformation space, calculate energies, heats of formation and strain energies, determine equilibrium and transition-state geometries, and evaluate normal-mode vibrational frequencies.

TITAN can run on Pentium-based PCs running Microsoft Windows 95, 98, or NT with at least 64 MB of RAM and SVGA graphics.

Schrödinger, Inc. Partners with COMPAQ in Product Development

Schrödinger, Inc. (*www.schrodinger-.com*) and Compaq Computer Corp. (*www.compaq.com*) are collaborating on current and future product development on Alpha systems running Tru64UNIX. The goal of the collaboration is to optimize Schrödinger's software products' performance using Compaq's Alpha technology. The current collaboration projects include the parallel Jaguar product, and Maestro, the new GUI application that will be released by the end of 1999. Jaguar is an *ab initio* electronic structure software package that provides chemical accuracy for realistic systems in reasonable time. Jaguar is much more accurate than semiempirical methods and is much faster than other *ab initio* methods.

Tripos Releases Software for Locating Binding Sites in Macromolecules

Tripos, Inc. (*www.tripos.com*) has introduced SiteID™, a software program that provides analysis and visualization tools for identification of potential binding sites within biological targets. SiteID's methods enable it to identify ligand binding pockets in any macromolecule, and potential sites for protein-protein interactions. SiteID enables analyses needed to locate binding sites, including H-bonding, hydrophobicity, and solvent accessible surface. Communication between Tripos' Molecular Spreadsheet™ and SiteID's macromolecule display results in an interface that unites analysis with visualization and enables exploration of macromolecules. Once a binding pocket has been identified, SiteID can generate the files needed for docking ligands via FlexX™.

SiteID will be available in the SYBYL™ 6.6 release scheduled for late October, 1999. SYBYL is Tripos' complete computational tool kit for molecular design and analysis. SYBYL provides essential construction and analysis tools for both large and small molecular structures.

MDL and Synopsys Release DataCartridge for Oracle 8i

Both MDL (*www.mdli.com*) and Synopsys Scientific Systems Ltd. (*http:// www.synopsys.co.uk/*) separately announced that they will release a chemically intelligent data cartridge that will allow research organizations to store and access chemical structures inside Oracle 8i™ databases by Oracle Corporation (*www.oracle.com*). The DataCartridge module for Oracle 8i transforms chemistry into a datatype, searchable alongside numbers and text via standard SQL. Integration within Oracle means chemical structures and reactions can be stored, searched, and analyzed within a relational framework.

Companies will be able to migrate to the new technology while maintaining their investment in ISIS (Integrated Scientific Information System) or related MDL systems. ISIS is the most widely used platform for chemical information management, and the new chemistry data cartridge will enable researchers to use standard ISIS search techniques such as exact match, substructure, and similarity within Oracle repositories. Support is also provided for SMILES strings, Molfiles, and Rxnfiles.

With the chemistry data cartridge, flexible structure matching provides control over chemical structure searches with flexible switches that include tautomer, stereoisomer, salt forms, fragment counts, and others — all within a single database search. The chemistry data cartridge is expected to be released by December, 1999.

APPLICATION NOTE

RS³ Discovery for Excel

Sheila Ash, Senior Cheminformatics Consultant New Business Development, and Glenn Atter Ph.D., Software Engineer, Oxford Molecular Limited, The Medawar Centre, Oxford Science Park, Sandford on Thames, Oxon OX4 4GA UK, sash@oxmol.co.uk, gatter@oxmol.co.uk

Abstract

The development of RS³ Discovery for Excel has allowed Oxford Molecular, through collaborative development on site at Wyeth-Ayerst Research, to develop a database access tool that is in tune with the way chemists and biologists perform their research. Its component based architecture has potential reuse in any Visual Basic application that needs to access RS³ Discovery functionality. The product has been successfully rolled out to scientists at Wyeth-Ayerst Research and Oxford Molecular is developing a fully object oriented Java API to the RS³ Discovery server.

Introduction

This paper was presented as a New Product Review at the International Conference on Chemical Structures held in Noordvijkerhout, The Netherlands, in 1999. Its purpose is to describe both RS³ Discovery for Excel, a new product being launched by Oxford Molecular and the manner in which this was conceived, designed, and built.

Background on Existing Oxford Molecular Products

Central to Oxford Molecular's software solutions for research information management is its RS³ Discovery Server. This is an ORACLE server application that includes patented tech-

nology that transforms ORACLE into a high-performance structure search engine. Structure searching includes exact match, substructure, similarity, tautomer, and stereochemical searches.

Also included is the RS^3 Data Catalog, a metadata dictionary that exposes Oracle tables to client applications for configuring combined searching. The Data Catalog is generalized for maintaining lists of identifiers for all types of objects and has an API for retrieving data from Oracle that caused a query to match (QBE).

Additional applications, such as Diamond Discovery, annotate chemical databases by providing additional information about predicted physical properties, predicted toxicology, pharmacophores, and other compound-based descriptors. These increase the value of the corporate database by adding new information that may help to direct future experiments and decisions.

Thus, with functionally powered by RS^3 Discovery Server and its Data Catalog, researchers can query over any Oracle table within their company to which they have authorized access. Since most pharmaceutical research companies store their structures, observed and calculated data, biological, physical property data, etc. in Oracle databases, this mean that these are readily accessible through the scientists' interface of choice.

RS^3 Discovery Server is based on a flexible, client/server architecture to provide high database performance accessible to users running on a variety of clients. Some of these client interfaces, such as TSAR, are aimed at specific scientific user groups, for example, QSAR specialists, diversity analysts, etc. Others are aimed at a more general scientific audience, for example, DIVA for scientists look for trend and patterns in data sets. These clients are Data Catalog enabled so tht their users have immediate access to all corporate Oracle-based data.

Collaborative Development

A number of companies had indicated that Excel would be their scientists' interface of choice. It was an existing RS^3 Discovery customer, Wyeth-Ayerst Research, which became the collaborative partner for Oxford Molecular for the development of an Excel based querying and reporting client to the RS^3 Discovery server.

Oxford Molecular already had the experience of working collaboratively with Glaxo Wellcome scientists in the design, construction, and testing of DIVA. Collaborative development projects entailed Oxford Molecular staff working at a partner's site and alongside practicing scientists. Together, they determine requirements in mutually-agreed, focused areas. The scientists actively participate in the software definition, specification and testing. This environment is conducive to rapid prototyping and review, with scientists giving input into all stages of the project. This approach proves very beneficial to both partners – Oxford Molecular gets software which reflects real needs and the scientific partner has an application that fits well into its own discovery process.

Application Requirements

As in other companies, Wyeth-Ayerst Research scientists were very familiar with Excel. As an existing RS^3 Discovery user, Wyeth-Ayerst Research required an Excel-based querying and reporting client for their RS^3 system.

Specifically, for Wyeth-Ayerst Research, the project aimed to develop a data access system which allowed all Wyeth-Ayerst scientists access to their data. For Oxford Molecular, the project aimed to produce commercially viable software for querying and reporting structures and data from RS^3 Discovery empowered Oracle environments.

Wyeth-Ayerst Research had some specific requirements of the resultant data access system, namely,

1. As an existing RS^3 Discovery user, standard RS^3 functionality had to be included.

   • List Manager – the management of queries and lists is critical functionality within RS^3 Discovery and its associated applications. This enables users to conduct edit, execute, and delete operations and to perform Boolean list creation.

   • Query Editor – the creation, editing, and execution of structure and data queries over any Data Catalog exposed Oracle table.

   • List/Query Property Editor – this covers object security, query periodicity, query domains, etc. These requirements were consistent with Oxford Molecular's desire to build a client for RS^3.

2. The application needed to handle over the approximately 26,000 different searchable data fields at Wyeth-Ayerst Research. These included biological data, compound data, inventory data, etc. and are grouped into approximately 2,000 RS^3 Query Sources. This again was consistent with Oxford Molecular's general requirement to cater to the vast data structures typical of large pharmaceutical companies. Finding the appropriate query source within such a large set as this could have been difficult. However, a folder hierarchy similar to a file system was used. By using a hierarchy, it is possible to group the query sources in a way which is appropriate to the Discovery environment. An example of this type of hierarchy is to group biological assays into therapeutic areas and then into different projects areas.
3. Scientists needed to be able to see only those query sources that were available for a given compound and to be able to drill down and see all the data for a given compound in a given query source.
4. Scientists needed to be able to retrospectively modify a query.
5. Scientists needed to be able to check for new data entered into the underlying Oracle tables which satisfies a query and to update a previously created report to include and highlight this new data. In practice, this important feature allows a chemist to easily keep track of the assay results that are being generated by the biologists.
6. While both Excel 95 and Excel 97 were in use and would remain in use, the final system at Wyeth-Ayerst Research had to be accessible from other VB/VBA application frameworks such as Access 95 and Access 97.

Application Architecture

This application had to fully interact with RS^3 Discovery and run as an add-in to Excel 95. This meant that the RS^3 components, including the List Manager and the Query Editor had to be built within the Excel95 Visual Basic (VB) development framework. Unfortunately, this is impossible to do in Excel 95 because the VB framework only allows very simple components to be used within a dialog. Another mechanism had to be found to allow more complicated dialogs to be used within Excel 95. The solution was to use Microsoft's Component Object Model (COM) and to develop these more complex external dialogs using C++ and MFC.

This approach has several advantages. First, the development time for the Visual Basic RS^3 application was reduced by taking the RS^3 code out of the VB development system and into a set of COM objects. The COM library effectively becomes a dialog resource library, connecting to RS^3 Discovery for all common dialog components. VB writers can produce applications incorporating these dialogs from RS^3 Discovery. Other reporting tools, not based upon Excel, could be developed within VB. Finally, the COM library simplifies the development process by enabling work to proceed without PL*SQL knowledge.

However, limitations of the COM functionality in Excel 95 do impose some restrictions on the interaction between the Excel spreadsheet and the GUI components. However, at the time of starting this project, market research indicated that a sizable Excel 95 user base existed and would continue to exist for sometime.

Figure 1 shows a schematic diagram of the architecture used in the RS^3 Discovery for Excel application. The advantage of this system is that a developer is never restricted to one level of access. For example, if there is not a requirement for a fully functional Query Editor then the one provided by the GUI components could be considered too complicated. However, the developer could build their own simplified query editor by accessing the lower level RS^3 connectivity objects.

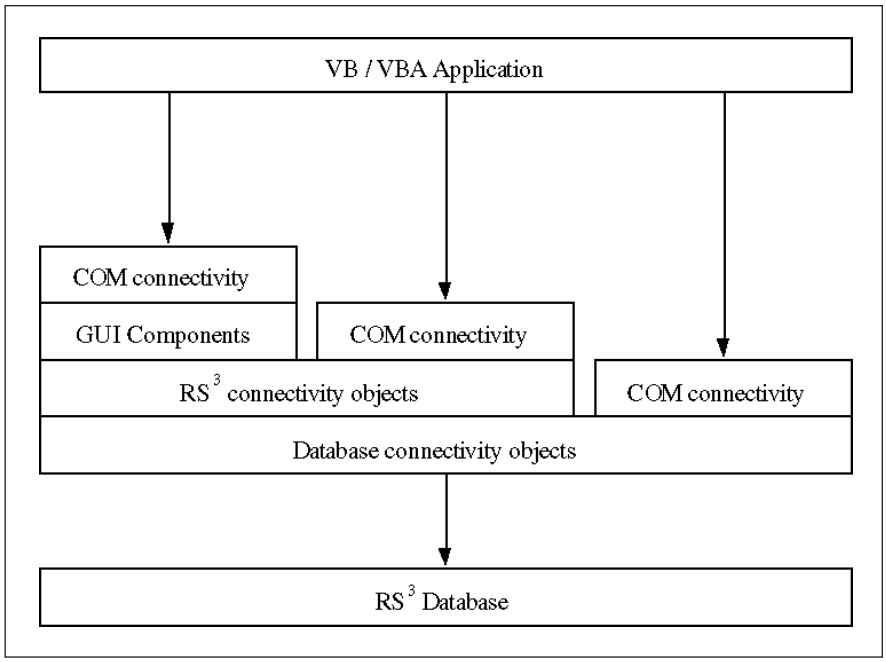

Figure 1. This figure shows how the RS3 Discovery for Excel system is structured. From the diagram, there are three main layers: a database connectivity layer to provide database access for the system, a set of RS3 object to facilitate access to RS3, and a GUI layer which provides the high-level GUI components such as the List Manager and Query Editor. It is important to note that the VB/VBA application has access to all three layers and as such it is not limited to a specific layer, it can always drop down a layer and provide its own higher-level system.

The Product

Today, all three Wyeth-Ayerst Research sites are using RS^3 Discovery for Excel as their main RS^3 Discovery client. An inhouse training program is underway and the system is in everyday use by staff. It is the preferred data access tool for scientists when a quick query is required against the database and when they need to report multiple compounds and data.

Full details on RS^3 Discovery for Excel readers are available at *http://www.oxmol.com/prods/rs3excel/* or *http://www.oxmol.co.uk/prods/rs 3excel/*.

The Future

The success of RS^3 Discovery for Excel has encouraged Wyeth-Ayerst Research to extend the approach to the forms-based method of reporting which they are now building. This manner of report is seen as the more appropriate when the scientist needs to report lots of data or all data for a single compound. From Oxford Molecular's viewpoint, the project has also been a success in that we are launching this new product with a proven easy to use and familiar Excel interface enabling scientists to rapidly query and report data.

For the future, Oxford Molecular is developing a fully object-oriented Java API to RS^3 Discovery comprising RS^3 foundation classes, Java classes, and COM objects for use in application development by Oxford Molecular and its customers.

BOOK REVIEWS

Molecular Modelling on the PC

By Matthew F. Schlecht (DuPont Agricultural Products, Delaware). Wiley-VCH, New York and Chichester. 1998. xviii + 763 pp. US$125.00 hardcover. ISBN: 0-471-18567-1.

One of the most significant trends in computational chemistry in the last 15 years is the increasing affordability and power of desktop computing. Calculations that used to require a distant

mainframe or supermini computer now can be easily performed in one's own office. Desktop computing is attractive because the calculations can be done at the convenience of the individual, not at the convenience of an information technology bureaucracy. Three factors have paved the way for this new age of desktop modelling: (1) less expensive personal computers with greatly increased memory and speed, (2) relatively inexpensive molecular modelling software, and (3) dissemination of information about how to do modelling. This book addresses this third point in a very effective way.

The book is a combination of a thorough tutorial into molecular modelling and an impressive user's manual for Serena Software's PCMODEL program. Serena is a company founded and run by Drs. Kevin Gilbert and Joe Gajewski of Indiana University at Bloomington in the hazy hills and hollows of southern Indiana. Despite the company's diminutive size, it has been able to compete effectively with the Goliaths of the molecular modelling software business. A bibliometric survey of software usage (Boyd, D.B. Molecular modelling software in use: Publication trends. In: *Reviews in computational chemistry*, Lipkowitz, K.B., and Boyd, D.B., Eds., VCH Publishers, New York, 1997, 6:317-354) found that PCMODEL had become one of the most widely used and cited molecular modelling programs. Hence the focus of this book on this program is entirely appropriate.

Various definitions of molecular modelling have been offered over the years. These definitions have ranged from the very narrow "empirical computation of conformational energy, combined with molecular graphics" to the very broad "anything which is done on a computer to depict, describe, or evaluate any aspect of the properties or structure of a molecule" (see Trost, B.M. *Science* 1985, 227, 908-916; Pensak, D.A. *Pure Appl. Chem*. 1989, 61:601-603). Dr. Schlecht offers his own definition: "use of graphical, mathematical, or physical representations to help understand and predict the properties of molecules." A book to cover such a broad definition would be monumental indeed. However, the author quickly tells the reader for purposes of this book the scope will be confined to "the application of empirical force fields to calculating molecular structure." The book beautifully fulfills this pragmatic mission.

The book begins with an enlightening introduction to force fields and modelling including historical perspectives for the newcomer. Next the book sets the stage with descriptions of hardware and software and explanations of input and output of data. A hefty chapter, which could be a book in itself, then describes essentially all the force fields of current interest. The last chapter gives examples of applications, ranging from conformational searches, to docking, to modelling inorganics. A useful appendix includes an explanation of units, a list of PCMODEL commands, a list of atom types in PCMODEL, and a lengthy glossary of terms in molecular modelling. A phenomenal 993 references back up the hefty contents of the book. A supplemental diskette attached to the book contains PCMODEL exercises and URLs relevant to the molecular modeller.

To write a book like this is not easy. Some points the author might want to consider if a second edition is produced are as follows. Whereas the index is substantial, it is mainly a subject index. The names of a few cited authors are interspersed in the list of subjects, but there is no full index of cited authors. Some readers will notice that the name of Professor Corwin Hansch, the famous father of QSAR, is misspelled more than once in the book. Whereas the word "energy" is one that has many different meanings to computational chemists, no definition is offered in the glossary under this word, although the various kinds of energy (free energy, van der Waals energy, and so on) are defined. Steric energy, as used by Allinger in the context of the MM2, MM3, and MM4 force fields, is not defined in the glossary. The free energy equations given with the definition of enthalpy and entropy have an incorrect sign in front of the TS term. However, these are minor points and will not interfere with the reader/student learning a great deal from the book in its present form.

In a spot check to see if the MM3 bond stretching term was presented correctly (rather than with the misprint in Allinger's original 1989 *J. Am. Chem. Soc*. paper), the book gives the equation correctly. Such attention to detail is reassuring.

The book came out in 1998 after 10 years in gestation. The contents are quite up-to-date. It is easy to be enthusiastic about this book. The author has done a remarkable job. The book targets a strong market demand for tutorials in molecular modelling on a personal computer. At only US$0.16 per page, this book is a bargain and should find a wide audience among users of PCMODEL and other molecular modelling packages. The book will help newcomers who want to learn how molecular modelling works and how to perform molecular modelling calculations properly. The book will advance the field of computational chemistry by broadening the base of scientists who use modelling in their research.

Donald B. Boyd, Ph.D., Department of Chemistry, Indiana University-Purdue University at Indianapolis (IUPUI), 402 North Blackford Street, Indianapolis, IN 46202-3274, boyd@chem.iupui edu

Encyclopedia of Computational Chemistry

Edited by Paul von Ragué Schleyer (University of Georgia, Athens, Emeritus, University of Erlangen-Nürnberg, Erlangen, Germany), Editor-in-Chief. John Wiley & Sons, New York. 1998. xxix + 3429 pp. US$3150.00 hardcover. ISBN: 0-471-96588-X.

The *Encyclopedia of Computational Chemistry* is a five-volume set with articles and definitions, and descriptions of topics and software covering most areas of computational chemistry. Editors Paul von Ragué Schleyer, Norman L. Allinger (Georgia), Tim Clark (Erlangen-Nürnberg), Johann Gasteiger (Erlangen-Nürnberg), Peter A. Kollman (San Francisco), and Henry F. Schaefer III (Georgia) have undertaken the monumental challenge of generating under one title the definitions and explanations of a very large and multifaceted area of science. With the

hard work of associate editor and project coordinator, Peter R. Schreiner (Göttingen/Georgia), the encyclopedia efficiently collates entries from about 450 contributing authors who are among the prominent leaders in the field of computational chemistry. The final product reflects the care and deep commitment made by the editors and contributors in their effort to generate a first-rate encyclopedia. In my opinion, they have succeeded in their endeavor, and I highly recommend this encyclopedia.

The work looks and feels like an encyclopedia: it consists of five large (8.5 by 11 inches) books containing over 3,400 pages, weighing about 30 pounds total, printed on heavy duty paper that will ensure its durability. These Wiley books are aesthetic to the hand as well as to the eye; the easy-to-read print, good use of headings and subheadings, and the judicious selection of color all contribute to the luxurious "feel" of the books.

In the Preface, Editor-in-Chief Schleyer publicizes the many other influential books and journals with which he and his co-editors have been associated. Among these is Allinger's *Journal of Computational Chemistry*, which "aptly named the emerging chemical discipline" in 1980. Schleyer also writes that *Reviews in Computational Chemistry* "merits special mention. (Its) many instructive contributions complement the chapters in ECC." Schleyer observes that he and his co-editors, except for Kollman, are part of the "Erlangen–Georgia axis."

My assessment of content is based on reading (1) selected entries about which I feel I am knowledgeable and (2) selected entries on topics I knew nothing about. The first check let me evaluate (partly) the technical merits of the contents. The second check let me determine if I was able to learn about new topics without being overwhelmed by jargon or technical details and to find suitable references so that I could further pursue a topic if I wished. As a third check, I asked other people in my department to look up topics in which they were interested.

My first reading was on Molecular Mechanics. This entry was a bit problematic; rather than being a self-contained description of the method, it defined what molecular mechanics is and then indicated where else in the encyclopedia one should go to learn about it. Confusing the issue is that the immediately following entry is about the more narrow topic of molecular mechanics treatments of conjugated systems. Further digging revealed other information about molecular mechanics. According to the index, molecular mechanics is briefly described in the article on Drug Design and in articles scattered though four of the five volumes. What one needs to know about molecular mechanics is thus in the encyclopedia but not quite where one would expect to find it.

The next two entries I evaluated were Conformational Analysis and Cambridge Structural Database. The conformational analysis description consists of three entries: Part 1 by Kolossváry and Guida, Part 2 by Vajda, and Part 3 by Eliel. I give all these entries a strong positive review. Although there exists some redundancy between these articles, each entry provides a different perspective, and the integration of the three entries into the encyclopedia is done well. All three entries are well written, concise, technically superb, and contain many leading references. The entry on the Cambridge Structural Database (CSD) by Allen and Hoy was likewise technically superb. It provides a history of CSD, what the CSD contains, what the software can do and then provides some research applications, recent developments, and what developments will appear in the near future. I scanned several other entries on topics I am familiar with, and I found them to be technically sound, having a concise but not terse writing style, and having references to key papers and reviews.

My next assessment was to read an entry or two about topics I knew little about to see how easily I could grasp the fundamental concepts being described. The first topic was on Comparative Molecular Field Analysis (CoMFA) written by Kubinyi. The author put its need into perspective, described the methodology, and presented applications. Major sections cover: series design and test set selection; pharmacophore hypotheses and alignment; box, grid size, and 3D field calculations; derivation and validation of 3D QSAR models; and some practical problems. Not only did I become aware of what I could do with CoMFA, I was also taught about pitfalls to avoid and given pointers and hints about how to be successful. Furthermore, I was provided with ample citations that would allow me to further pursue this topic. Like several other topics I read about, I was able to quickly get the gist of what the computational method was used for and how it worked.

I also took the opportunity to invite some colleagues to look up a topic or two of interest to them to and tell me what they thought about what they read. The messages about the quality of the entries was mixed. The first topic was "polarizability." A meager, completely unsatisfactory definition and explanation was provided. Indeed, one general problem I found with the encyclopedia was the inconsistency of topical entries. The one paragraph on polarizability, a topic I deem to be of major importance in computational chemistry, was followed by five pages on the topic of polymer brushes! Several other topics that colleagues looked up included AutoDock, logP, and lipophilicity. The first two were covered only briefly, and the third was not in the subject index. However, a student could find about lipophilicity by reading the entry on Octanol/Water Partition Coefficients, but would a student have the savvy to look it up there? In a further test, one of our graduate students looked up Graph Theory and Electronegativity. In both instances, she was impressed with the completeness of the descriptions as well as the utility of the encyclopedia for general student use. Both entries were readable, guiding her throughout the intricacies of these topics, and both were well illustrated having high quality pictures that made clear the conceptual ideas being described.

The fifth volume of this five-volume set includes a section with brief promotional descriptions of some relevant programs; these are written by software developers and vendors who were invited to contribute such information. This volume also has an

index of contributing authors and a subject index. The subject index is 55 pages long and will help the reader track down dispersed topics. Also, at the end of many of the articles and definitions is a list of related articles to help the reader delve further. An index of cited authors is lacking.

Overall, this encyclopedia is very useful. It was found by me, my colleagues, and students to be useful in our research as well as in the educational missions we have at a university. This set of books will thus enrich the field by instructing the general user and facilitating investigations as the editors say. The physical and aesthetic attributes of the books are superior. There is good use of color as needed. The contents are excellent in spite of some weak entries and unevenness in topics covered. It was wonderful to have a review copy of these books, but at a price of about 90 (U.S.) cents per page, actual purchases will most likely be by libraries. I can recommend the encyclopedia highly to any university or company interested in the many concepts and tools of computational chemistry.

Kenny B. Lipkowitz, Ph.D., Department of Chemistry, Indiana University-Purdue University at Indianapolis, 402 North Blackford Street, Indianapolis, IN, 46202-3274,. lipkowitz@chem.iupui.edu

PAPER ALERTS

Structure-based Drug Design: Combinatorial Chemistry and Molecular Modelling

Comb. Chem. High Throughput Screen., 1999, 2(4):211-221

D.L. Kirkpatrick, S. Watson, S. Ulhaq, ProlX Pharmaceuticals, Inc., Pittsburgh, PA

This article describes the shift in drug discovery efforts to utilize the rapid synthetic procedures of combinatorial chemistry along with rational library design. The many available computational methods which examine both the receptor structure and the ultimate pharmacore complementarity offer novel approaches whereby chemists may not only discover new lead compounds but also design virtual libraries for screening prior to the synthetic stage. Current methods of library generation are reviewed, highlighting docking procedures useful in both the discovery and optimization stages of drug development. Specific examples illustrate approaches to the use of docking, including a description of the development of inhibitors to the human A3 adenosine receptor and HIV-1 protease, and the evaluation of the activity of novel inhibitors of the redox regulator protein, human thioredoxin.

Virtual Combinatorial Syntheses and Computational Screening of New Potential Anti-Herpes Compounds

J. Med. Chem., 1999, 42(17):3308-3314

J.V. de Julian-Ortiz, J. Galvez, C. Munoz-Collado, R. Garcia-Domenech, C. Gimeno-Cardona , Unidad de Investigacion de Diseno de Farmacos y Conectividad Molecular, Facultat de Farmacia, and Departamento de Microbiologia, Hospital Clinico Universitario, Universitat de Valencia, Spain, julian@colom.combios.es

The authors use an in vitro assay to determine the activity of new anti-HSV-1 chemical structures designed and selected by virtual combinatorial chemical synthesis and computational screening. Two databases of building blocks, one of carbonyl fragments and the other containing both substituted phenoxy and phenylamino fragments, are used to build a virtual library. Compounds in the virtual library were computationally screened, and compounds selected by the authors' mathematical model as active were synthesized and tested. An antiviral activity model was employed consisting of a "tandem" of four linear functions of topological graph-theoretical descriptors. Chemical structures were selected as active if they satisfied all discriminant equations in that model. Five new structures were selected and tested. All selections demonstrated activity, and three showed appreciable anti-HSV-1 activity, with IC(50) values of 0.9 microM. The same model has been applied to a database of known compounds and has identified anti-herpes activity in the following compounds: 3,5-dimethyl-4-nitroisoxazole, nitrofurantoin, 1-(pyrrolidinocarbonylmethyl)piperazine, nebularine, cordycepin, adipic acid, thymidine, alpha-thymidine, inosine, 2, 4-diamino-6-(hydroxymethyl) pteridine, 7-(carboxymethoxy)-4-methylcoumarin, 5-methylcytidine.

Computational Modelling of the Rate Limiting Step in Low Molecular Weight Protein Tyrosine Phosphatase

FEBS Lett., 1999, 456(2):301-305

K. Kolmodin, J. Aqvist, Department of Cell and Molecular Biology, Uppsala University, Biomedical Center, Sweden

The reaction catalyzed by the protein tyrosine phosphatases (PTPs) includes hydrolysis of the phosphoenzyme intermediate as the second and limiting step. The cysteinyl phosphate thioester bond is cleaved by nucleophilic displacement where an active site water molecule attacks the phosporus atom. The authors observe the energetics of this reaction using the empirical valence bond method in combination with the MD and FEP simulations, and modelled the reactions of the wild-type as well as the D129A and C17S mutants. An alternative reaction mechanism is proposed for the D129A mutant. The calculated activation barriers are in all cases in accord with experimental reaction rates. These results along with earlier computational and experimental work offer a good picture of the complete reaction mechanism in many PTPs. The key role of the structurally invariant signature motif in stabilizing a double-negative charge is shown by its control of the energetics of both transition states and the reaction intermediate.

Distinguishing Between Sequentially and Nonsequentially-folded Proteins: Implications for Folding and Misfolding

Protein Sci., 1999, 8(8):1591-1604

C.J. Tsai, J.V. Maizel Jr, R. Nussinov , Laboratory of Experimental and Computational Biology, NCI-FCRDC, Frederick, MD 21702

The authors offer an algorithm for distinguishing sequentially from nonsequentially-folding proteins. Recent experiments have suggest that most of the proteins that are synthesized in the eukaryotic cell may fold sequentially. This proposed folding mechanism in vivo is particularly advantageous to the organism. Furthermore, the probability that a sequentially folding protein will misfold is reduced significantly in the absence of chaperones. The authors attempt to devise a procedure that will differentiate between the two types of folding patterns. Footprints of sequential folding may be found in structures where consecutive fragments of the chain interact with each other. In these cases, the folding itself may be seen as less complex. While on the other hand, higher folding complexity suggests that some portion of the polypeptide backbone folds back upon itself to form three-dimensional (3D) interactions with noncontiguous portion(s) of the chain. Consequently, the mechanism of folding of the molecule is observed via analysis of its complexity, i.e., through the 3D interactions formed by contiguous segments on the polypeptide chain. The structure is computationally spliced into consecutively interacting fragments, by cutting it into compact hydrophobic folding units or into a set of hypothetical, transient, highly populated, contiguous fragments. In sequential folding, successive building blocks interact with each other from the amino to the carboxy terminus. Results of the parsing differentiate between sequentially vs. nonsequentially-folded chains. A simple, sequentially-folded protein should be less error prone and fold faster than a protein with a complex folding pattern as in the funnel free energy landscape theory, while a protein that follows the mechanism of sequential folding encounters smoother free energy barriers.

A Database Method for Automated Map Interpretation in Protein Crystallography

Proteins, 1999, 36(4):526-541

D.J. Diller, M.R. Redinbo, E. Pohl, W.G. Hol, Department of Biological Structure, University of Washington, Seattle, WA 98195

Many new protein structures contain folds that are related to those already observed. While developing a computer program that can accurately position, in electron density maps, large protein domains with large structural deviations, redundancy in protein folds could be used during a protein structure determination. A computational procedure, Database Assisted Density Interpretation (DADI), was developed to aid in the building of models in protein crystallography and to assist in interpreting electron density maps. The authors describe the initial tests of the DADI procedure using a small database of protein domains. Their approach is to work first with entire domains, then with the secondary structure elements of these domains, and finally with individual residues of the secondary structure elements via Monte Carlo, "chopping," and "clipping" procedures, respectively. Their first test case is a traceable 3.2 Å multiple isomorphous replacement with anomalous scattering (MIRAS) electron density map of a human topoisomerase I-DNA complex. The second test case uses poor electron density for the third domain of the diphtheria toxin repressor resulting from a molecular replacement solution with the first two domains. The DADI procedure successfully found a large portion of the protein backbone with very few errors. In the first case, nearly 45% of the backbone and more than 80% of the secondary structure was placed automatically. In the second case, nearly 50% of the third domain was automatically detected. In addition, in both cases more than 75% of the beta sheet secondary structure was found automatically by the DADI procedure. The procedures offers avenues for exploiting the current protein structures for the determination of future structures.

Structural Principles Governing Domain Motions in Proteins

Proteins, 1999, 36(4):425-435

S. Hayward , BIOSON Research Institute, Laboratory of Biophysical Chemistry, University of Groningen, Groningen, The Netherlands, steve@chem.rug.nl

Twenty-four proteins for which two or more X-ray conformers are known were analyzed using a recently developed method to reveal structural principles that govern domain motions in proteins. In all 24 cases, the domain motion is a rotation about a physical axis created through local interactions both covalent and noncovalent. In many cases, two or more mechanical hinges separated in space create a stable hinge axis for precise control of the domain closure. In a significant number of cases, the terminal regions of α-helices and β-sheets were found to act as mechanical hinges. In some cases, the two terminal regions of neighboring strands of a single β-sheet can create a hinge axis, as can the two termini of a single α-helix. These two structures have been termed the "double-hinged β-sheet" and "double-hinged α-helix," respectively. Another way of constructing a hinge is to use a flexible loop that attaches one domain to another and through which the effective hinge axis passes. Noncovalent interactions between segments remote along the polypeptide chain can also form hinges. α-helices that preserve their hydrogen bonding structure when bent have been found to behave as mechanical hinges. The author suggests that these α-helices act as a store of elastic energy that drives the closing of domains for rapid capture of the substrate. If, as this study suggests, the repertoire of possible interdomain structures is quite limited, the dynamic behavior of proteins could soon be

predicted using bioinformatics techniques.

Binding of Buried Structural Water Increases the Flexibility of Proteins

Proc. Natl. Acad. Sci. USA, 1999 96(17):9613-9615

S. Fischer, C.S. Verma, Computational and Structural Chemistry, Hoffmann-La Roche/Pharma Research, CH-4070 Basel, Switzerland,. fischer@iwr.uni-heidelberg.de

Water deeply buried in proteins is considered to be an integral part of the folded structure, with these structural water molecules making strong H bonds with polar groups of the surrounding protein and as such are believed to tighten the protein matrix. Through computational analysis of the binding of a buried water molecule to bovine pancreatic trypsin inhibitor, the authors unexpectedly found that the protein actually becomes more flexible, as revealed by an increase in the vibrational entropy. They posit that this effect must be common in proteins, because the large entropic cost of immobilizing a single water molecule ($-T\Delta S = 20.6$ kcal/mol) can only be partly compensated by water-protein interactions, even when they are nearly perfect, leaving no room for a further decrease in entropy from protein tightening. Changes in protein flexibility are important for the prediction of ligand binding affinities.

In Silico Design for Protein Stabilization

Curr. Opin. Biotechnol. 1999 10(4):387-390

B.I. Dahiyat , Xencor Inc., 2585 Nina Street, Pasadena, CA 91107, baz@xencor.com

Building upon newly developed computational protein design methods that have recently been applied to problems in protein stabilization, the author describes the experimental testing of several sequence-design strategies which has demonstrated that a wide range of protein structures can be stabilized. The primary advantage of computational design is the vast number of sequences that can be rapidly screened in the search for an optimal design, far exceeding non-computational methods. This feature allows for the discovery of very large changes in protein properties.

Computational EST Database Analysis Identifies a Novel Member of the Neuropoietic Cytokine Family

Bioem. Biophys. Res. Commun., 1999, 262(1):132-138

Y. Shi, W. Wang, P.A. Yourey, S. Gohari, D. Zukauskas, J. Zhang, S. Ruben, R.F. Alderson, Department of Molecular Biology, Department of Pharmacology, Department of Protein Expression, Human Genome Sciences, Inc., 9410 Key West Avenue, Rockville, MD 20850

A novel member of the neuropoietic cytokine family was cloned and the protein expressed and characterized. In an attempt to identify novel secreted proteins, an algorithm incorporating neural network algorithms was applied to a large EST database. A full-length clone was identified that has a single open reading frame of 225 amino acids. This new cytokine is most homologous to cardiotrophin-1, having a similarity and an identity of 46 and 29%, respectively, and therefore has been named cardiotrophin-like cytokine (CLC). It is highly expressed in spleen and peripheral leukocytes. Purified recombinant CLC induced the activation of NFκB and SRE reporter constructs in the TF-1, U937, and M1 cell lines. In addition, the signal transduction pathway for CLC was characterized in the neuroblastoma cell line SK-N-MC and found to involve tyrosine phosphorylation of gp130 and STAT-1.

Molecular Dynamics Simulations of P450 BM3 — Examination of Substrate-induced Conformational Change

J. Biomol. Struct. Dyn., 1999, 6:1189-1203

Y.T. Chang, G.H. Loew, Molecular Research Institute, Mountain View, CA 94043-2316, chang@purisima.molres.org

Bacterial cytochrome P450 BM3 is one of only five isozymes of over 400 metabolizing heme proteins with a known crystal structure and only one of two where both substrate-free and substrate-bound forms are known. P450 BM3 has a similar function and substrates to the mammalian P450 4A subfamily. The extent to which P450 BM3 undergoes a conformational change that substrate-free P450 BM3 undergoes upon binding of a typical fatty acid substrate, palmitoleic acid, has been the subject of recent experimental studies. Direct examination of the substrate-free (pdb2hpd.ent and pdb2bmh.ent) and substrate-bound (pdb1fag.ent) forms do not provide a clear answer to this question, because the two substrate-free monomers reported in the crystal structures have significantly different conformations from each other, one with a more open substrate-access channel than the other. There is no way to tell to which form the substrate binds, so the effect of substrate binding cannot be deduced directly from comparisons of the experimental substrate-bound and substrate-free forms. MD simulations were performed for each of the two substrate-free forms found in the asymmetric unit of the X-ray structure and for the two corresponding substrate-bound forms, constructed by docking palmitloeic acid into each of them. Comparisons of the results showed that palmitoleic acid binding had little effect on the conformation of the more closed form of P450 BM3. In the more open form, substrate induced a closing of the entrance to the substrate-binding channel. The MD averaged structure of these two complexes obtained from docking of pamitoleic acid into the two asymmetric units of the substrate-free form were also compared to that obtained starting with the X-ray structure of the substrate-bound form. If the substrate induces conformational changes in P450 BM3, the mouth of the substrate-access channel first closes down in response to the presence of the substrate, then rotation of the F-G domain

further optimizes the P450 BM3-substrate interaction.

Rational Combinatorial Library Design. 3. Simulated Annealing Guided Evaluation (Sage) of Molecular Diversity: A Novel Computational Tool for Universal Library Design and Database Mining

J. Chem. Inf. Comput. Sci., 1999, 39(4):738-746

W. Zheng, S.J. Cho, C.L. Waller, A. Tropsha , Laboratory for Molecular Modeling, School of Pharmacy, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599-7360

A method for molecular diversity sampling called SAGE (simulated annealing guided evaluation of molecular diversity) is described. Compounds in chemical databases or virtual combinatorial libraries are conventionally represented as points in multidimensional descriptor space. The SAGE algorithm selects a desired number of optimally diverse compounds from a database. The diversity of a subset of compounds is measured by a specially designed diversity function, the most diverse subset is selected using Simulated Annealing (SA). Application of SAGE to two simulated data sets of randomly distributed points in two-dimensional space afforded diverse and representative selection as judged by visual inspection. SAGE was also applied to two other simulated data sets with points distributed among many clusters. SAGE sampling covered significantly more clusters than the random sampling. SAGE was compared with random sampling in terms of hit rates. When the percentage of active points was low, the hit rates obtained by SAGE were always higher than those obtained by random sampling. When the percentage of active compounds was high, the performance of SAGE depended upon the data structure. In all cases, SAGE performed better than random sampling when using cluster hit rates as the criterion.

Genome-based Structural Biology

Prog. Biophys. Mol. Biol., 1999, 72(1):1-17

D. Frishman, H.W. Mewes, GSF-Forschungszentrum fuer Umwelt und Gesundheit, Munich Information Center for Protein Sequences, am Max-Planck-Institut fur Biochemie, Martinsried, Germany, frishman@mips.biochem.mpg.de

This review describes computational approaches and results in protein structure analysis stemming from the availability of complete genomes.

On the Importance of a Methyl Group in Beta-Lactamase Evolution: Free Energy Profiles and Molecular Modelling

Biochemistry, 1999, 38:10499-10510

N.J. Bernstein, R.F. Pratt , Department of Chemistry, Wesleyan University, Middletown, CT 06459

β-Lactam antibiotics are generally thought to inhibit their target enzymes, the bacterial cell wall-synthesizing DD-peptidases, because of their resemblance to D-alanyl-D-alanine peptides. Although a favorable conformation of D-alanine does structurally resemble the β-lactams with respect to backbone conformation, a significant difference is the presence of a D-methyl substituent on the penultimate alanine residue of the cell wall peptide. A classical β-lactam antibiotic has a hydrogen in the corresponding position. In the process of evolution of a β-lactamase from a DD-peptidase, it seems likely that this D-methyl group would be selected against, to ensure that the former enzyme would hydrolyze β-lactams rather than peptides. In this paper, the effect of the penultimate D-alanine residue (as opposed to a glycine residue) has been examined in peptide substrates of a present-day DD-peptidase and a β-lactamase. The peptides N-(phenylacetyl)-D-alanyl-D-phenylalanine and N-(phenylacetyl) glycyl-D-phenylalanine were used as a test pair against the DD-peptidase of Streptomyces R61 and the class C β-lactamase of *Enterobacter cloacae* P99. Aminolysis by D-phenylalanine of a cognate pair of depsipeptides was studied to measure partitioning of the acyl-enzyme intermediate. Free energy-reaction coordinate diagrams were constructed for turnover of both peptides by both enzymes. The D-methyl group is preferred over hydrogen by the DD-peptidase at all stages of catalysis (acyl-enzyme and acylation and deacylation transition states), whereas the β-lactamase selects against the D-methyl group only at the peptide acylation transition state. Evolution of the methyl group by the β-lactamase has apparently occurred. Models of the acylation tetrahedral intermediates were made. A methyl group pocket on the DD-peptidase was identified. Problems posed by larger substituents on the penultimate residue of the peptide, and in particular by the heterocyclic substituent present in a bicyclic β-lactam, were analyzed. Models support the proposed importance of the penultimate D-alanine in β-lactamase evolution.

LearnCoil-VMF: Computational Evidence for Coiled-Coil-like Motifs in Many Viral Membrane-fusion Proteins

J. Mol .Biol., 1999, 290(5):1031-1041

M. Singh, B. Berger, P.S. Kim, Department of Biology, 9 Cambridge Center, Cambridge, MA 02142

The coiled-coil motif occurs in several viral membrane-fusion proteins, including HIV-1 gp41 and influenza virus hemagglutinin. The LearnCoil-VMF program identifies coiled-coil-like regions in viral membrane-fusion proteins. The authors report detailed sequence analyses of coiled-coil-like regions in retrovirus, paramyxovirus, and filovirus membrane-fusion proteins. Sequence analyses of these proteins outside their putative coiled-coil domains illustrate some structural differences. The coiled-coil-like regions detected by LearnCoil-VMF provide further evidence that the three-stranded coiled coil is a common motif found

in many diverse viral membrane-fusion proteins. The abundance and structural conservation of this motif, suggests it is critical for viral-cellular membrane fusion. (The program is available at *http://web.wi.mit.edu/kim.)*

Rapid Calculation of Polar Molecular Surface Area and its Application to the Prediction of Transport Phenomena. 2. Prediction of Blood-Brain Barrier Penetration

J. Pharm. Sci., 1999, 88(8):815-821

D.E . Clark, Rhone-Poulenc Rorer, Dagenham Research Centre, Rainham Road South, Dagenham, Essex, RM10 7XS, United Kingdom, david-e.clark@rp-rorer.ac.uk

A QSAR model for the prediction of log BB from a set of 55 diverse organic compounds is described here. The model contains two variables: polar surface area (PSA) and calculated logP. It predictions log BB for large compound sets. Performance of this QSAR on two test sets taken from the literature is illustrated.

Detecting Protein Function and Protein-Protein Interactions from Genome Sequences

Science, 1999, 285(5428):751-753

E.M. Marcotte, M. Pellegrini, H.L. Ng, D.W. Rice, T.O. Yeates, D. Eisenberg, UCLA-Department of Energy Laboratory of Structural Biology and Molecular Medicine, University of California at Los Angeles, Los Angeles, CA 90095-1570

A method is proposed for predicting protein interactions from genome sequences on the basis of the observation that some pairs of interacting proteins have homologs in another organism in a single protein chain. Searching sequences from many genomes revealed 6809 proposed protein-protein interactions in *Escherichia coli* and 45,502 in yeast. Many members of these pairs are functionally related. Some proteins have links to several other proteins and may represent functional interactions such as complexes or pathways. There is a Database of Interacting Proteins with experimentally confirmed interacting pairs at *http://dip.doe-mbi.ucla.edu.*

Analysis of Fas-ligand Interactions Using a Molecular Model of the Receptor-ligand Interface

J. Comput. Aided. Mol. Des., 1999, 13(4):409-418

J. Bajorath, MDS Panlabs, Computational Chemistry & Informatics, Bothell, WA 98011-8805

A molecular model of the complex between Fas and its ligand is presented. The model helps understanding of the location and putative effects of site-specific mutations, interactions at the Fas-FasL interface, and identification of contact residues. Regions in Fas and its ligand which could not be predicted with confidence were omitted from the model to ensure accuracy of the analysis. It was possible to map four of five N-linked glycosylation sites in Fas and FasL and to study 10 of 11 residues identified by mutagenesis as important for binding. The predicted structure of the Fas-FasL interface was consistent with the experimental evidence for binding by these residues. Five residues were identified and predicted to contribute to binding via electrostatic interactions. The model provides a basis to understand the role of Fas and FasL residues for binding.

Approximate Solvent-Accessible Surface Areas from Tetrahedrally-directed Neighbor Densities

Biopolymers, 1999, 50(4):373-380

J. Weiser, P.S. Shenkin,W.C. Still , Anterio Consult & Research GmbH, Augustaanlage 26, D-68165 Mannheim, Germany, jweiser@anterio.com

A fast analytical formula has been derived for the calculation of approximate atomic and SASA, as well as the first and second derivatives of these quantities. Extending the work of Stouten et al. (Mol. Sim., 1993, 10:97-120), and the authors' own (J. Comp. Chem., 1999, 20:586-596), the method makes use of a Gaussian function to calculate the neighbor density in four tetrahedral directions, sometimes twice with different orientations. SASA and first and second derivatives of the heavy atoms of penicillopepsin are computed in 0.13 and 0.23s, respectively, on an SGI R10000/194 MHz processor, faster than other algorithms. Based on a parameterization set of nineteen compounds of different sizes (11-4346 atoms) and chemical classes, the method exhibits relative errors of 0.2 to 12.6% for total molecular surface areas and average absolute atomic surface area deviations of 1.7 to 3.6 $Å^2$.

PATENT ALERTS

Production of Enzymes Having Desired Activities by Mutagenesis

Patent Number: US5939250
Inventor: Short, Jay
Assignee: Diversa Corporation, San Diego CA, USA

Abstract

The patent includes claims covering the evolution of naturally occurring heterogeneous populations of molecules by a wide range of methods to generate proteins or enzymes having desired characteristics. This patent protects screening and evolving heterogeneous populations of DNA. Diversa has discovered and characterized over 1,000 enzymes from these recombinant libraries. Evolution techniques have been applied to these and other molecules to produce highly optimized proteins and small molecules of interest. This patent includes coverage for screening a heterogenous DNA population which has been exposed to mutagenesis towards production of a specified enzyme or protein, irrespective of the mutagenesis technique employed.

Antibacterial Drugs, Methods for Their Design and Methods for Their Use

Patent Number: WO9943338 A1
Inventor: Glinski, Guennadi
Assignee: Metastat, Inc.,
San Diego, CA, USA

Abstract
Antibacterial drugs and general methods of design and use of the inhibitors of bacterial growth and antibacterial drugs are described. The methods are based on design and application of the compounds blocking the assembly and function of DNA-dependent RNA-polymerase (RNAP) by targeting protein-protein contact site(s) and nucleic acid binding site(s) of at least one protein component of RNAP and thereby inhibiting the subunit-subunit interactions essential for RNAP assembly and function. Specific examples of the antibacterial drugs, proposed method of antibacterial drug design, and use based on the inhibition of protein-protein and protein nucleic acid interactions are presented.

Arylthiadiazole Derivative and Antiviral Agent Containing the Same

Patent Number: US5948916
Inventors: Ijichi; Katsushi, Shigeta; Shiro, Baba; Masanori, Fujiwara; Masatoshi, Yokota; Tomoyuki, Takayama; Hiromitsu, Sakai; Shin-ichiro, Hanasaki; Yasuaki, Ide; Teruhiko, Watanabe; Hiroyuki, Katsuura; Kimio
Assignee: Rational Drug Design Labs, Japan

Abstract
Novel arylthiadiazole derivatives and salts thereof useful for preventing and treating human viral infection and a novel virucide which contains the arylthiadiazole derivative or a salt thereof are provided. N,N-dimethyl [3-(3-(amino-2,6-dichlorophenyl)-1,2,5-thiadiazol-4-yl] carbamate or its salt, and virucide containing the same as an effective component.

Method and Apparatus for Evaluating Molecular Similarity

Patent Number: WO9944055 A1
Inventor: Nicholls, Anthony,
Assignee: Openeye Scientific, Inc.
Santa Fe, NM, USA

Abstract
This patent describes several techniques for characterizing molecules based on the shapes of their field properties such as steric or electrostatic. The minimal distance between two molecular fields is used as a shape-based metric, independent of the underlying chemical structure, and a shape space description matrix (**D**) is generated. A metric matrix **C** is then generated from matrix **D**. The matrix **C** is then diagonalized to find the eigenvalues. The next step is to find coordinates of structures in N-dimensional shape space. Coordinates are discarded to reduce dimensionality from N to M such that the distances in matrix **D** are still reproduced within a certain tolerance. The result is an **M** dimensional shape space (where (M<N) and with a certain tolerance for a chosen field property. The patent shows how these attributes can be used in creating, characterizing, and searching databases of molecules based on field similarity. In particular, they allow searches of a database in sub-linear time. Describing a way to automatically break molecules into a series of fragments by using an ellipsoidal Gaussian decomposition extends the utility of this approach. Not only can these fragments then be analyzed by the shape metric technique described above, but the parameters of the decomposition themselves can also be used to further organize and search databases. The ellipsoidal method can also be used to describe binding or active sites on macromolecules, providing a template for searching for complementary molecules in a database. The most immediate application of these techniques is to drug design.

L-4' Arabinofuranonucleoside Compound and Medicine Composition Comprising the Same

Patent Number: WO9943690 A1
Inventors: Sato, Hiroshi; Yoshimura, Yuichi; Ashida, Noriyuki; Sudo, Kenji; Yokota, Tomoyuki

Abstract
A L-4'-thioarabinofuranonucleoside compound, a 5 membered sulfur heterocycle, with an extra cyclic substituent that can be a nucleic acid base selected from among pyrimidine, purine, azapurine, and deazapurine, each of which may be substituted with a halogen atom, an alkyl group, a haloalkyl group, an alkenyl group, a haloalkenyl group, an alkinyl group, an amino group, an alkylamino group, a hydroxyl group, a hydroxyamino group, an aminoxy group, an alkoxy group, a mercapto group, an alkylmercapto group, an aryl group, an aryloxy group, or a cyano group. These compounds are useful as antivirals especially against hepatitis.

Ligand Screening and Design by X-ray Crystallography

Patent Number: WO9945389 A2
Inventors: Nienaber, Vicki; Greer, Jonathan; Abad-Zapatero, Celerino; Norbeck, Daniel
Assignee: Abbott Laboratories

Abstract
X-ray crystallography can be used to screen compounds for their ability to bind the target biomolecule. The method includes obtaining a crystal of a target biomolecule; exposing the crystal to test samples; and obtaining diffraction data to determine whether a ligand/receptor complex is formed. The target is exposed to test compounds by either co-crystallizing a biomolecule in the presence of test samples or soaking the crystal in a solution of test samples. Also included in this invention is forming a biomolecule crystal having an easily accessible active site by co-crystallizing the target with a ligand which is subsequently degraded.

Condensed Heterocyclic Compounds as Anti-inflammatory and Immunomodulatory Agents

Patent Number: WO9945926 A1
Inventors: Shannon, Patrick; Eichholtz, Thomas; Linstead, David; Masdin,

Philip; Skinner, Richard
Assignee: University College Cardiff Consulting Ltd., Cardiff, UK

Abstract
The invention describes compounds for use as an immunomodulatory or anti-inflammatory drugs or for treatment of a therapeutic indication in which inhibition of dehydro-orate dehydrogenase (DHODH) is beneficial.

A Human Vanilloid Receptor-Like Cation Channel WO9946377 A2

Patent Number: WO9946377A9
Inventors: Partiseti, Michel; Renard, Stéphane
Assignee: Sanofi-Synthelabo, France

Abstract
Human vanilloid receptor-like cation channel (hVRCC) polypeptides and polynucleotides and methods for producing them by recombinant techniques are disclosed. Also disclosed are methods for using hVRCC polypeptides and polynucleotides in the design of protocols for the treatment of acute and chronic inflammation, acute and chronic pain, brain diseases, abnormal proliferation and cancer, ulcer, autoimmune diseases, control of viscera innervated by the dorsal root ganglia neurons (for instance control of bladder function or dysfunction), to mimic or antagonize effect of endogenous neurotransmitters and hormones, to inhibit graft rejection by promoting immunosuppression, among others, and diagnostic assays for such conditions.

Small Molecule Inhibition of RNA/Ligand Binding

Patent Number: US5935776
Inventors: Green, Michael; Zapp, Maria
Assignee: University of Massachusetts Medical Center, Worcester, MA USA

Abstract
This patent discloses a method for the inhibition of binding of a ligand to an RNA, the inhibition being mediated by 2,5-Bis[4-(2-N,N-dimethylaminopropylamidino)phenyl]furan that binds to the RNA, thereby inhibiting ligand binding.

Method and System for Protein Modelling

Patent Number: US5958672
Inventors: Srinivasan, Subhashini; Sudarsanam, Padmanaban
Assignee: Immunex Corp., Seattle WA USA

Abstract
This patent describes a protein modelling method. The modelling is based upon the structure of a template protein and a sequence alignment. For each residue in the model, when the template protein has an amino acid aligned with the amino acid of the model protein, the position of the backbone model is established based on the position of a topologically equivalent backbone atom in the aligned amino acid of the template protein. The modelling of a variable region of the model protein is based on a collection of psi and phi angle values for amino acid pairs in a family of proteins. In a further embodiment, phi and psi angles are classified according to a tetramer of adjacent residues and filtered based on a most probable conformation of portions of the variable region of the model protein.

UPCOMING MEETINGS

October 1999

Identification of Novel Bioactive Compounds

October 20-22, 1999
Brussels, Belgium

Contact:
Amy L. Dasch
Cambridge Healthtech Institute
1037 Chestnut Street
Newton Upper Falls, MA 02464 USA
Tel.: 617-630-1352
Fax: 617-630-1325
adasch@healthtech.com
www.healthtech.com/conference/mhe/mhe.htm

Joint 55th Southwest/ 15th Rocky Mountain Regional ACS Meeting

October 21-23, 1999
El Paso, TX USA

Contact:
K. Pannell
Chemistry Dept.
University of Texas
El Paso, TX 79968-0513
Tel.: 915-747-5796
Fax: 915-747-5748
kpannel@utep.edu

Eleventh International Chemical Information Conference and Exhibition

October 25-28, 1999
Annecy, France

Contact:
Infonortics Ltd.
15 Market Place
Tetbury, Glos GL8 8DD UK
Tel.: +44 1666 505 772
Fax: +44 1666 505 774
chemical@infonortics.com
www.infonortics.com

EuroLabAutomation '99

October 25-29, 1999
Novotel Hammersmith, London, UK

Contact:
EuroLabAutomation
4 Rivercourt, Trinity Street
Oxford, OX1 1TQ UK
eurolab99@aol.org
www.eurolabautomation.org

34th Midwest Regional ACS Meeting

October 27-29, 1999
Quincy, IL USA

Contact:
H.D. Wohlers
Truman State University
Science Hall
100 East Normal
Kirksville, MO 63501-4435
Tel.: 816-785-4625
Fax: 816-785-4045
wohlers@truman.edu

November 1999

CHIPS to HITS '99: Harnessing the Power of Microtechnology

November 2-5, 1999
The Claremont Resort and Spa
41 Tunnel Road

Berkeley, CA 94705 USA

Contact:

225 Turnpike Road
Southborough, MA 01772-1749
Tel.: 508-481-6400
Fax: 508-481-7911
reg@ibcusa.com
www.chipstohits.com

2nd Annual NMR Technologies: Development and Applications for Drug Discovery

November 4 -5, 1999
Sheraton Inner Harbor Hotel
Baltimore, MD USA

Major advances in NMR technologies continue to enable dynamic, molecular structure studies leading to a greater understanding of some of the mechanisms of various diseases as well as targeted therapeutic drug design and discovery programs. New biomedical and materials sciences are fueling the development of high-field NMR spectroscopy as a more accessible tool for researchers. Researchers are continuing to develop new ways to share their NMR resources more efficiently and creatively, primarily over the Internet. Driving the interest in faster access to SAR analyses is genome research. Major emphasis will be placed on the application of NMR to drug discovery processes this year, including case studies and other examples from large pharma and biotech organizations. Researchers involved in solid-state and solution NMR spectroscopy, structural biology, materials and polymer sciences, and genomic sciences, among others, are encouraged to submit a title and brief summary describing their work for consideration of presentation.

Contact:

Carol A. Ghatak
Conference Director
Cambridge Healthtech Institute
1037 Chestnut Street
Newton Upper Falls, MA 02464
Tel.: 617-630-1348
Fax: 617-630-1325
cghatak@healthtech.com
www.healthtech.com

European MDL Software Users Group Meeting

November 8-10, 1999
Montreux, Switzerland

Contact:

Brigitte Vögelin
MDL Information Systems AG
Gewerbestrasse 12, CH-4123
Allschwil 2, Switzerland
Tel.: +41 61 486 88 75
Fax: +41 61 486 88 89
brigitte.voeglin@mdli.com
www.mdli.com

ChiraSource '99

November 15-19, 1999
Philadelphia, PA USA

Contact:

Catalyst Group, PO Box 637,
Spring House, PA 19477
Tel.: 215 628 4447
Fax: 215 628 2267
cnf@catalystgrp.com
www.catalystgrp.com

Protein Structure

November 15-16, 1999
The Capital Hilton,
Washington, D.C. USA

Topics include: advances in structure determination, computed protein structure, protein functional analysis, and implications for target selection and drug discovery.

Contact:

Cambridge Healthtech Institute
1037 Chestnut Street
Newton Upper Falls, MA 02464
Tel.: 617-630-1300
Fax: 617-630-1325
chi@healthtech.com
www.healthtech.com

3rd International Conference for Chemical Information Users

November 16-17, 1999
Manchester Conference Centre,
UMIST, Manchester, UK

The conference is sponsored by the Royal Society of Chemistry Chemical Information Group and the Chemical Structure Association. The themes of the conference will be:

- Electronic Journals and Document Delivery
- Education and Industry: Information Training for Chemists
- New Methods of Representing Chemical Structures on the Web

Contact:

Dot Snow
Royal Society of Chemistry
Chemical Information Group
Dot.Snow@agrevo.com
chemweb.com/conference/
manchester/manchester_general.html

3rd Annual Conference on Computational Genomics

November 18-21, 1999
Baltimore MD USA

TIGR
Tel.: 301-610-5959
Fax: 301-838-0229

Combinatorial Chemicals & Catalysis Optimization Conference & Exhibition

November 18-19, 1999
The Ritz Carlton Hotel
Philadelphia, PA USA

Contact:

Ms. Connie Scarcia-Haverstick
Conference Manager
The Catalyst Group
P.O. Box 637
Spring House PA 19477
Tel.: 215 628 4447
Fax: 215 628 2267
cnf@catalystgrp.com

8th Asian Chemical Congress

November 21-24, 1999
Taipei, Taiwan

Contact:

8th Asian Chemical Congress
Secretariat
Applied Chemistry Division
Union Chemical Laboratories
ITRI
321 Kuang Fu Rd., Section 2
Hsinchu, Taiwan 300
Tel.: +886 3 573 2535
Fax: +886 3 573 2348
740123@ucl.itri.org.tw

Research Informatics for Drug Discovery

November 22-23, 1999
Park Hyatt, Philadelphia, PA USA

The current complexity of storing and manipulating data from such areas as: bioinformatics, high-throughput screening results, early ADME screens, and large multivariate clinical trial data will be addressed in this conference. Parallels between the pharmaceutical industry's needs and information management technology from existing industries such as banking will be explored. Additionally, the advantages and draw-backs of data standardization within companies as well as within the industry at large will be discussed. Both information users and IT professionals will find this conference informative and provocative.

Contact:
Amy L. Dasch, Conference Director
Cambridge Healthtech Institute
Tel.: 617-630-1352
Fax: 617-630-1325
adasch@healthtech.com
www.healthtech.com

Solutions for Scientists Symposium

November 29-30, 1999
Olympia, London, UK

Topic Areas Covered:
- Miniaturization/Automation
- Chromatography Data Handling
- LIMS
- Implementation of Laboratory Computer Systems
- Data Warehousing/Storage
- Molecular Modelling
- Spectroscopy in Process and Quality Control
- Atomic Spectroscopy
- Genomics/Proteomics
- Bioinformatics/Combinatorial Chemistry

Contact:
Simon Roberts
Advanstar Communications
Sealand Road
Chester, Cheshire CH1 4RN UK
Tel.: +44 (0) 1244393161
Fax: +44 (0) 1244370011
sroberts@advanstar.com
www.s3event.com

December 1999

Online Information '99

December 7-9, 1999
National Hall & Olympia 2
London, UK

Online Information '99 will provide a forum for the world's information professionals and end-users interested in information strategy and implementation. This is an opportunity to discuss future developments in information access, content, and the impact of technology across all industry sectors. Online Information is the largest information industry event in the world.

Contact:
Learned Information Europe Ltd.
Woodside, Hinksey Hill
Oxford OX1 5BE UK
Tel.: +44 (0) 1865 388000
Fax: +44 (0) 1865 736354
exhibitions@learned.co.uk
www.online-information.co.uk

Innovative Techniques for ADME

December 13-14, 1999
US Grant Hotel
326 Broadway
San Diego, CA 92101 USA

More than 70% of all drug candidates fail in clinical trials because of poor ADME properties. This huge obstacle in drug discovery and development can increase the time and expense of new drug development. Screening for human metabolic stability, absorption, drug-drug interactions, and other ADME properties early in drug discovery allows for the selection of compounds with optimal properties for further development. A number of new ADME assays and screens are rapidly being developed, yet scientists are facing challenges in keeping up with these innovations and implementing them with maximum efficiency.

This hands-on program is designed to provide a working knowledge of drug metabolism and its applications in the ADME laboratory. The seminar reviews the newest philosophies in drug metabolism and the approaches and techniques necessary for increasing drug candidate survivability earlier in the discovery phase. Computational applications and tandem technologies are also discussed along with the latest *in vivo* dosing techniques. Details are provided for:

- Developing 3D membrane transport protein structures
- Analyzing the involvement of Cytochrome P450 in metabolism
- Testing drug candidates for delivery early in discovery
- Increasing rapid screening *in vivo*
- Applying in vitro drug-drug interaction screens and assays
- Estimating which Cytochrome P450 isoforms metabolize your candidate early in development
- Exploring permeability (MDCK, CaCo2, BBB) screens
- Choosing cassette dosing techniques
- Optimizing drug potency and pharmacokinetics

Contact:
Institute for International Research
708 Third Ave, 2nd Floor
New York, NY 10017-4103
Tel.: 888 670-8200 (USA)
Tel.: 941 951-7885 (Outside USA)
Fax: 941 365-2507
info@iir-ny.com
www.iir-ny.com

January 2000

Pacific Symposium on Biocomputing 2000

January 4-9, 2000
Sheraton Waikiki Hotel
2255 Kalakaua Ave.
Honolulu, HI USA

PSB is a forum for the presentation of work in databases, algorithms, interfaces, visualization, modelling and other computational methods, as applied to biological problems, with emphasis on applications in data-rich areas of molecular biology.

Contact:
Kevin Lauderdale
PSB Coordinator Section on Medical Informatics
SUMC, MSOB X-215
Stanford CA 94305-5479
Tel.: 650-725-0659
Fax: 650-725-7944
psb@cgl.ucsf.edu
www.cgl.ucsf.edu/psb

Bioinformatics Europe

January 20-21, 2000
SwissHotel Zurich
Zurich, Switzerland

Contact:
Jennifer Laakso
Cambridge Healthtech Institute
1037 Chestnut Street
Newton Upper Falls, MA 02464
USA
Tel.: 617-630-1300
Fax: 617-630-1325
chi@healthtech.com
www.healthtech.com/

February 2000

Exploiting Molecular Diversity — Refining Small Molecule Libraries

February 7-9, 2000
Hyatt Regency La Jolla
3777 La Jolla Village Drive
San Diego CA 92122 USA

Contact:
Cambridge Healthtech Institute
1037 Chestnut Street
Newton Upper Falls, MA 02464
USA
Tel.: 617-630-1300
Fax: 617-630-1325
chi@healthtech.com
www.healthtech.com/conference/mld2000/mld2000.htm

High-throughput Organic Synthesis

February 10-11, 2000
Hyatt Regency La Jolla
3777 La Jolla Village Drive
San Diego, CA 92122 USA

Contact:
Jennifer Laakso
Cambridge Healthtech Institute
1037 Chestnut Street
Newton Upper Falls, MA 02464
Tel.: 617-630-1300
Fax: 617-630-1325
chi@healthtech.com
www.healthtech.com/

2000 Charleston Conference: Innovative Techniques for Lead Discovery and Development

February 28- March 1, 2000
Isle of Palms, SC USA

This annual conference examines the latest research techniques in bioinformatics, high throughput screening, combinatorial chemistry, and predictive toxicology/ADME. Topics currently identified for possible coverage at the 2000 Conference will include:

- Genomics
- Combinatorial Chemistry/ Medicinal Chemistry
- High Throughput Screening
- ADME/Toxicology
- Virtual Chemistry
- Virtual Screening
- QSAR
- New Approaches to Informatics
- Computational Approaches to Solids

Contact:
Merry Ambos
Network Science Corporation
412 Carolina Blvd.
Isle of Palms, SC 29451-2113
Tel.: 843 886-5008
Fax: 843 886-5924
mambos@netsci.org
www.netsci.org

March 2000

Second Log P Symposium. Lipophilicity in Drug Disposition: Practical and Computational Approaches to Molecular Properties Related to Drug Permeation, Disposition, and Metabolism

March 5-9, 2000
University of Lausanne
Switzerland

The Symposium will focus on the determination, computation, and interpretation of lipophilicity and related molecular properties as factors and predictors of drug permeation, disposition, and metabolism. Strong introductory emphasis will be given to the biological and pharmacokinetic background.

In experimental themes, particular attention will be paid to the lipophilicity profiles of ionized compounds, to lipophilicity measurements in anisotropic media (liposomes/water, IAM columns), and to permeability across artificial membranes. The relevance of these parameters to pharmacokinetic properties will be examined. Computational themes will comprise lipophilicity and H-bonding fields and their interest in docking strategies and structure-permeation relations.

Throughout the Symposium, lipophilicity and related parameters will be contemplated from a dual perspective, their interpretation in terms of recognition forces, and their value in screening, lead optimization, and drug candidate selection.

Scientific and Organizing Committee

- Prof. Bernard Testa, Chair (School of Pharmacy, University of Lausanne)
- Prof. Gerd Folkers (Dept of Pharmacy, Swiss Federal Institute of Technology - Zurich)
- Dr. Pierre-Alain Carrupt (School of Pharmacy, University of Lausanne)
- Mrs. Nicole Matter (Lausanne Tourism)
- Dr. Joachim Mayer (School of Pharmacy, University of Lausanne)
- Dr. Marianne Reist (School of Pharmacy, University of Lausanne)
- Dr. Han van de Waterbeemd (Pfizer Central Research, Sandwich, UK)

Contact:
Register Online at
http://ictsg10.unil.ch/logp2000

GENOMICS: New Discoveries and Commercial Developments

March 29-31, 2000
Churchill College
Cambridge
Cambridgeshire, UK

Genomics is poised to play an increasing role in medicine and agriculture. In the next few years, the complete sequences of hundreds of genomes will be available and recent developments have indicated that the human genome sequence will be finished ahead of the 2005 target date. These

sequence data will be accompanied with a vast amount of additional information including results of comparative genomics, RNA and protein expression profiles, functional and pathway information and tools for molecular and cell biology.

How is the wealth of genome information going to be interpreted? How is this information going to be managed and shared? What impact has genomics had in the commercial environment and what are the future opportunities for exploitation of genomic information? This conference will provide a broad, in depth overview of new discoveries and commercial developments in genomics and will provide a framework for timely discussion of key issues.

The program has been designed to appeal to scientists working in the area of genomics, to those developing enabling technologies and to managers considering broader application of genomics in the Pharmaceutical and Agribusiness industries.

Sessions/Topics to Include:

- Structural genomics
  - Progress in plant and animal sequencing
  - Future challenges
- Gene function and expression
  - Expression profiling
  - Antisense
  - Gene regulation
  - Pathways
- Bioinformatics
  - Data mining and visualization
  - Genome mapping
- Applications and opportunities
  - Gene expression
  - Cells as factories
  - Pharmacogenetics
- The way forward?
  - Commercial Developments
  - Impact on the industries

Contact:

Sonia Walter
Society of Chemical Industry
14/15 Belgrave Square
London SW1X 8PS UK
Tel.: +44 (0) 171 235 3681
Fax: + 44 (0) 171 235 7743
soniaw@chemind.demon.co.uk
http://sci.mond.org/

April 2000

RECOMB2000, The Fourth International Conference on Computational Molecular Biology

April 8-11, 2000
Tokyo Big Sight
Tokyo, Japan

Contact:

Human Genome Center
University of Tokyo, Japan
recomb2000@ims.u-tokyo.ac.jp
http://recomb2000.ims.u-tokyo.ac.jp

Royal Society of Chemistry Annual Congress

April 17-20, 2000
UMIST
Bridgewater Hall, Manchester, UK

The conference is not intended to duplicate specialist meetings, but rather to attempt to portray chemistry from the widest possible perspective in a way which is relevant to all RSC members, to promote collaboration, and encourage new areas of research spanning the classical divisions of chemistry.

A key theme of this millennial Annual Conference is to look to the future, whether this be exploring chemistry at the interface with biology, developing clean and sustainable processes to reduce the environmental impact of our industry, or the quest for nanoscale devices.

Contact:

Nicole Morgan
2000 Annual Conference
Conferences Department
The Royal Society of Chemistry
Burlington House
Piccadilly , London W1V 0BN UK
Tel.: + 44 171 437 8656
Fax: + 44 171 734 1227
conferences@rsc.org
www.rsc.org/lap/confs/annconf2000.htm

May 2000

Second Indo-U.S. Workshop on Mathematical Chemistry

May 30-June 3, 2000
University of Minnesota Duluth
Duluth, MN USA

The workshop will bring together leading researchers in the field of mathematical and computational chemistry. The results of latest research and applications of mathematical and computational chemistry in drug discovery, environmental toxicology, quantitative structure-activity relationships (QSAR), quantitative molecular similarity analysis (QMSA), chemoinformatics and bioinformatics will be discussed. Dilip K. Sinha, Visva Bharati University (India) and Subhash C. Basak, Natural Resources Research Institute, University of Minnesota Duluth, are the co-chairs.

Contact:

Subhash C. Basak
Workshop Co-chairperson
Natural Resources Research Institute
University of Minnesota Duluth
5013 Miller Trunk Highway
Duluth, MN 55811
Tel.: 218 720-4230
Fax: 218 720-4328
sbasak@wyle.nrri.umn.edu
http://wyle.nrri.umn.edu/IndoUS2/

June 2000

10th International Congress of Quantum Chemistry

June 5-10, 2000
Palais de l'Europe
8 avenue Boyer
Menton, 06500
France

Organizing committee:

- Ernest R. Davidson
- Nicholas C. Handy
- Sigrid D. Peyerimhoff
- Alberte Pullman
- Jean-Louis Rivail
- Bjorn Roos

Contact:

Xth ICQC 2000
Laboratoire de Chimie Theorique
Universite Henri Poincare-Nancy I
B. P. 239
F- 54506 Vandoeuvre Les Nancy
Cedex France
icqc@lctn.u-nancy.fr
www.lctn.u-nancy.fr/ICQC

August 2000

ISMB2000: 8th International Conference on Intelligent Systems for Molecular Biology

August 20-23, 2000
University of California, San Diego
La Jolla, CA USA

This Conference provides a general forum for disseminating the latest developments in bioinformatics. ISMB is a multidisciplinary conference that brings together scientists from computer science, molecular biology, mathematics, and statistics. Its scope includes the development and application of advanced computational methods for biological problems.

ISMB 2000 will place special emphasis on knowledge discovery from the modeling and simulation of complex biological systems. This includes interpretation of large-scale gene expression data, whole genome comparative analysis, mathematical modelling of biochemical pathways, and interpretation of large macromolecular assembles using data at different resolutions.

Relevant Computational Techniques Include:

- machine learning
- pattern recognition
- knowledge representation
- databases, combinatorics
- stochastic modelling
- string and graph algorithms
- linguistic methods
- robotics
- constraint satisfaction
- parallel computation

Biological Areas of Interest Include:

- molecular structure
- genomics
- molecular sequence analysis
- evolution and phylogenetics
- molecular interactions
- metabolic pathways
- regulatory networks
- developmental control
- general molecular biology

Contact:

Nancy Jensen
Tel.: 858 534 5039
ismb00@sdsc.edu.
http://ismb00.sdsc.edu