A PROTOTYPE MOLECULAR INTERACTIVE COLLABORATIVE ENVIRONMENT (MICE)

P. BOURNE[1,2,4,a], M. GRIBSKOV[1,3], G. JOHNSON[1], J. MORELAND[1], S. WAVRA[1], H. WEISSIG[1,2]

[1]*San Diego Supercomputer Center*
P.O. Box 85608, San Diego CA 92186 USA

[2]*Department of Pharmacology*
[3]*Department of Biology*
University of California, San Diego
9500 Gilman Dr., La Jolla CA 92093 USA

[4]*The Burnham Institute*
10901 North Torrey Pines Road, La Jolla CA 92037 USA

Illustrations of macromolecular structure in the scientific literature contain a high level of semantic content through which the authors convey, among other features, the biological function of that macromolecule. We refer to these illustrations as molecular scenes. Such scenes, if available electronically, are not readily accessible for further interactive interrogation. The basic PDB format does not retain features of the scene; formats like PostScript™ retain the scene but are not interactive; and the many formats used by individual graphics programs, while capable of reproducing the scene, are neither interchangeable nor can they be stored in a database and queried for features of the scene. MICE defines a Molecular Scene Description Language (MSDL) which allows scenes to be stored in a relational database (a molecular scene gallery) and queried. Scenes retrieved from the gallery are rendered in Virtual Reality Modeling Language (VRML) and currently displayed in WebView, a VRML browser modified to support the Virtual Reality Behavior System (VRBS) protocol. VRBS provides communication between multiple client browsers, each capable of manipulating the scene. This level of collaboration works well over standard Internet connections and holds promise for collaborative research at a distance and distance learning. Further, via VRBS, the VRML world can be used as a visual cue to trigger an application such as a remote MEME search. MICE is very much work in progress. Current work seeks to replace WebView with Netscape, Cosmoplayer™, a standard VRML plug-in, and a Java-based console. The console consists of a generic kernel suitable for multiple collaborative applications and additional application-specific controls. Further details of the MICE project are available at *http://mice.sdsc.edu*.

1 Introduction

Rapid increases in the availability of three-dimensional structural data, the availability of inexpensive graphics workstations (including PCs), and the pervasive role of the Internet in molecular biology research and education, have created a need for better tools that can describe and manipulate molecular scenes. A molecular scene is defined here as a renderable image of a molecule with associated descriptive annotation. The

[a] To whom correspondence should be addressed.

scene may include several different types of molecular depiction (ball and stick, ribbons, CPK, surfaces, etc.) and specify coloring schemes, or other visual cues, that help describe the properties of the molecule. Further, a molecular scene should be able to trigger external applications relevant to the further exploration of the scene. Software tools that access molecular scenes should enable researchers and students to take full advantage of the high information content of visual information, as well as effectively collaborate and discuss the visual data, even when working from geographically distant locations.

In this paper we describe a prototype set of such tools, referred to as Molecular Interactive Collaborative Environment (MICE). MICE has three components:

1. A Molecular Scene Description Language (MSDL);
2. A molecular scene gallery;
3. Software for collaborative access to molecular scenes.

These components combine to provide a platform independent, inexpensive, freely available, collaborative environment for the study of molecules. It must be emphasized that MICE is very much work in progress. Component 3., has been implemented and tested, component 1. is under development, and work on component 2. has yet to begin.

1.1 *Potential Uses of MICE*

In designing MICE we set out to solve a series of problems from our own research and teaching experience. A subset of those problems are given here, along with the MICE component (italics) that best addresses each problem.

- Research collaborations on topics such as protein modeling and drug design increasingly involve diverse groups of investigators who are geographically remote. *A collaborative molecular visualization environment might potentially facilitate research at a distance*.
- While conducting a molecular modeling course in a room equipped with overhead projection from a single Macintosh server and with the students working from Macintosh clients all linked to the server and the Internet, a student asks a question about the structure versus sequence homology found in the light and heavy chains of deoxyhemoglobin. Having superimposed the two chains in 3-D to show the major insertion/deletion points additional questions arose about the molecular scene and then confusion along the lines "no I don’t mean that residue, but the one behind it and to the left." The learning process would have been simpler if the student could have taken control of the molecular scene via her computer screen and shown the whole class what she was pointing out. *A collaborative molecular visualization environment is potentially useful in teaching laboratories.*

- Scientific journals are increasingly available on line. However, the images presented in those papers are, for the most part, static or require specialized software for viewing. *MSDL provides sematically rich renderable images suitable for use in electronic publishing*.
- Again, from the molecular modeling course, a student asks the question "Can you show us other examples of this hydrogen bonding scheme [found in γ-crystallin] in other four-stranded Greek key motifs, annotated [donor and acceptor labeled and connected by dotted white lines] in the same way?" This question consists of two distinct parts: (i) a search for a common structure motif - a four-stranded Greek key; and (ii) a search for a specific type of annotation. While community attention, by way of Web-accessible databases, has focused on answering the first part, little attention has been paid to the ability to query components of a molecular scene. *A molecular scene gallery is a valuable community resource which can be queried for features of the image.*

1.2 Relationship of MICE to Other Work

The similarities and differences of MICE to other work is discussed for the three MICE components introduced above.

1.2.1 Molecular Scene Description Language (MSDL)

Most of the popular current generation macromolecular display programs provide an import/export facility. That is, the semantic content of a molecular scene can be saved and later restored. However, the saved scene is not interchangeable among graphics programs and hence not a general purpose exchange format. For example, a Rasmol script[24] or an Insight II PSV file[17], respectively, cannot be read by other graphics programs. Kinemage[22] has some of the features of MSDL in that it can be edited with a text editor or the Kinemage display program to add or change characteristics of the molecular scene, including the annotations. Ferrin and coworkers[13] have defined scene information as an extension of the PDB format. Finally, Brenner and Hubbard[8] have defined yet another molecular annotation format. None of the current molecular scene formats are interchangeable, rigorous, nor can any of them be loaded into a database and queried for characteristics of a molecular scene. MSDL has all these characteristics.

1.2.2 Molecular Scene Collaborative Software

Many groups are working on collaborative software which is, or could be, applied to biological macromolecules. A sample of these applications is given here for comparative purposes.

An Advanced Visualization System (AVS) module called SHARE_VIEWS (*ftp://ftp.iavsc.org/pub/AVS5/Module_Src/mappers/VIEW_SHARE*) has been developed by the North Carolina Supercomputing Center and allows up to 9 heterogeneous workstations to perform shared visualization. While the functionality of SHARE_VIEWS has some of the functionality of MICE (it shares static but not dynamic views), it has a major shortcoming. It is too slow to be a production level tool. A result of the constant polling that goes on between client and server and which consumes as much as 90% of the CPU on a low-end Silicon Graphics workstation.

EyeChem is a modular visualization environment developed by Henry Rzepa and colleagues[9]. The system is based on Silicon Graphics Iris Explorer and hence not portable to other platforms in common use in molecular biology, namely the PC and Mac. The MICE-like components are Eye2Eye, a collaborative environment, and the EyeChem VRML browser. Like MICE, these applications are in the early stage of development. Rzepa and colleagues are also embracing Java with the development of a console tool called MOzART, again in an early stage of development.

Video conferencing and shared whiteboards via the Internet (e.g., Intel Proshare[16], Silicon Graphics Inc. InPerson[27], respectively are capable of supporting scientific collaborative environments. However, they do not support scientific visualization at the level of sophistication defined by MICE. The same is true of Habañero[21] developed by the National Center for Supercomputing Applications (NCSA) and Tango[2] developed by the Northeast Parallel Architectures Center and Syracuse University. Both support collaborative environments, but either do not support 3-D visualization, or are not light-weight enough to run efficiently on a PC or Macintosh computer.

1.2.3 Molecular Scene Gallery

Several collections of molecular images already exist (e.g., Proteins Motions Database, *http://hyper.stanford.edu/~mbg/ProtMotDB*/ and SWISS-3DIMAGE, *http://expasy.hcuge.ch/sw3d/sw3d-top.html).* These are valuable resources, yet they can only be queried by basic features such as PDB code and compound name, and not by features of the molecular scene features. Further, they only contain static images, not detailed interactive molecular scenes.

Similarly, collections of macromolecular images based upon VRML have been developed (e.g., the MathMol library, *http://www.nyu.edu/pages/mathmol-/library/library.html* and the Jena Image Library of Biological macromolecules, *http://www.imb-jena.de/IMAGE.html)*. However, these libraries do not contain VRML displaying the detailed molecular scene description specified by MSDL. The MICE molecular gallery complements these resources.

2 Materials and Methods

A schematic showing the MICE design is given in Figure 1. Software components that have been prototyped at the time of writing (September, 1997) are shaded. The arrows indicate the flow of data in the format given in italics.

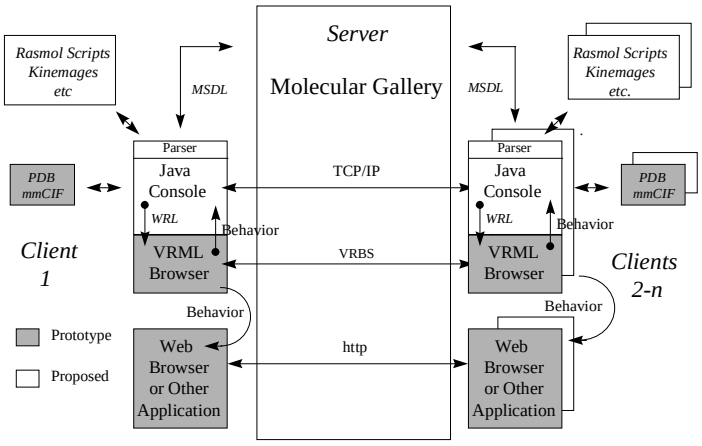

Figure 1: MICE Overview.

MICE implementation is discussed for each of the three components.

2.1 Molecular Scene Description Language (MSDL)

MSDL is based on STAR/CIF and makes use of a variety of existing software. The Crystallographic Information File (CIF) data representation, a subset of the Self-defining Text Archival and Retrieval (STAR) data representation[12] is the standard form of interchange for small molecule crystallographic data[15]. The extension covering macromolecules (mmCIF)[6] includes an extensive dictionary (over 3000 definitions) and dictionary definition language (DDL). All fields of the DDL, the mmCIF dictionary, and any mmCIF compliant data file are parsable and completely described in the dictionary. Hence, mmCIF is more comprehensive and rigorous than the PDB format. What is not covered in version 1.0 of the International Union of Crystallography (IUCr) approved mmCIF dictionary (nor by the PDB format) are definitions for a full description of the molecular scene. There is a provision for describing the orientation matrix with respect to the original atomic coordinate set and for describing fine details of secondary structure e.g., beta bulges, 3_{10} helices, but not the details of a specific view. For example, what atoms are included in the view, how

they are rendered, and how they are annotated. The MSDL dictionary, as an extension to the mmCIF dictionary, uses the full description of a structure as found in the mmCIF dictionary, but also provides a full description of the molecular scene. Once the dictionary is complete any molecular scene can be described using terms from the dictionary.

What makes this representation different to the another graphics formats described in Section 1.2.1? The answer lies in the data structure imposed by STAR/CIF and the DDL. STAR/CIF data items are grouped into a loop structure where all items in the loop belong to the same category. Certain data items are defined as obligatory if the loop is to be valid. The loop forms the basis of a relational table or a complex object and the obligatory data items the primary key, or object identifier, respectively. Any molecular scene described by MSDL can be loaded into a databases (including the molecular scene gallery) for subsequent query of features of the scene. It is not possible to do this with other graphics formats. Since MSDL is based on STAR/CIF basic parsing and data manipulation tools are available[7,11]. Details of the MSDL dictionary will be reported elsewhere.

2.2 Molecular Scene Collaborative Software

Consider, from the viewpoint of client 1, how the MICE prototype (shaded boxes) functions. A WRL (VRML file containing the molecular scene) is calculated in several steps starting from a PDB file. The WRL is then loaded from an Internet server into WebView, an SGI-specific VRML browser developed at the San Diego Supercomputer Center (SDSC)[20]. To date these WRLs only comprise color coded molecular ribbon representations with a few associated behaviors (see below) which are added by manually editing the WRL. Ribbons were chosen for prototyping since they were both descriptive and involve a large number of VRML primitives. Ribbons provided a significant test of the performance of VRML for these types of applications. The WebView browser supports the Virtual Reality Behavior System (VRBS)[18,19], a light-weight protocol and associated software. VRBS was originally developed to support behaviors, events triggered from the virtual world defined using VRML v1.0. VRML v2.0 contains behaviors part of the base specification, but VRBS is retained as an efficient means of communicating between remote VRML browsers as part of the collaborative environment. The VRBS protocol is not discussed in detail, additional details are available on-line[18,19]. It is sufficient to note that VRBS is string-based with no multi-byte binary data included, avoiding any machine dependency associated with byte ordering and floating point conversions. VRBS provides an efficient collaborative interaction between all extant computing platforms.

A key feature of the collaborative environment is synchronizing the activities of multiple users. In the MICE prototype, synchronization is handled by audio coordination via telephone or Internet and use of a single keystroke to gain and release control. An icon appears to indicate a user is in control. The icon is green if it is you in

control and red if someone else is in control. This is clearly inadequate. Users need to be able to add and delete themselves from a queue to take control, and control should be given automatically to the user at the top of the queue when the previous user relinquishes control. Synchronization is a key component of current software development efforts and will be controlled by the Java console described subsequently.

It has been demonstrated that the molecular scene can be divided (e.g., into enzyme and inhibitor) and that each component of the scene can be manipulated independently. This adds a new dimension to synchronizing the activities of multiple collaborating users, which is yet to be addressed.

Behaviors can be used to either update the molecular scene, or invoke an external application, the results of which appear in a standard Web browser (Figure 1). In the MICE prototype one behavior links color coding of various properties of the structure being displayed to various function keys. Thus, with a single keystroke, color coding for hydrophobicity, isotropic temperature factors, highlighting of the active site can be displayed. Another behavior links specific parts of the molecular scene to URLs which displayed Web pages in a separate Web browser when selected. A useful mechanism for detailed annotation of a region of structure. For example, it was possible to select a region of the structure of cAMP dependent protein kinase and update Web browsers on the various participating clients with multiple sequence alignments[27] of other members of the protein kinase family of enzymes for just that region of structure. Finally, a behavior initiated from a partial sequence found in the molecular scene was used to invoke a protein sequence motif search using MEME[14]. The application was run on a remote supercomputer with results returned to Web browsers on each client.

Current software development efforts logically separate the VRML browser, the Application Program Interface (API) between browser and console, and the Java console itself (Figure 1). To a MICE user this distinction is not apparent, however, to a software developer it means the VRML browser and API can be used with consoles developed for a variety of applications.

2.2.1 The VRML Browser

The existing WebView VRML browser is based on OpenGL and currently runs only on Silicon Graphics platforms. This browser is not generic enough for the intended audience. Current efforts are directed towards using Cosmoplayer™, a plug-in to the Netscape Navigator™ Web browser, and using Java beans to define the console to control the molecular scene. If this proves inadequate new libraries are emerging and will be used to build a freely available VRML browser from scratch. A notable development is Java/3D, a Java extension reported to have the functionality of OpenGL for 3-D applications. Whatever the development path, the goal is a VRML browser with the functionality of WebView, but platform independent. The mode of

navigation (e.g., fly-through, walk-through) of VRML browsers provides a fresh perspective on molecular rendering. A description of the capabilities of WebView is available at the URL *http://www.sdsc.edu/projects/vrml/tools/webview/help/-webview_running.html*.

2.2.2 The Java Console

The Java console logically encapsulates generic and MICE specific software components. The generic components are a standard set of control widgets used to interact with the VRML browser (e.g., sliders buttons) and are being developed for several unrelated applications. The MICE specific components of the Java console are illustrated in Figure 2 and their status described.

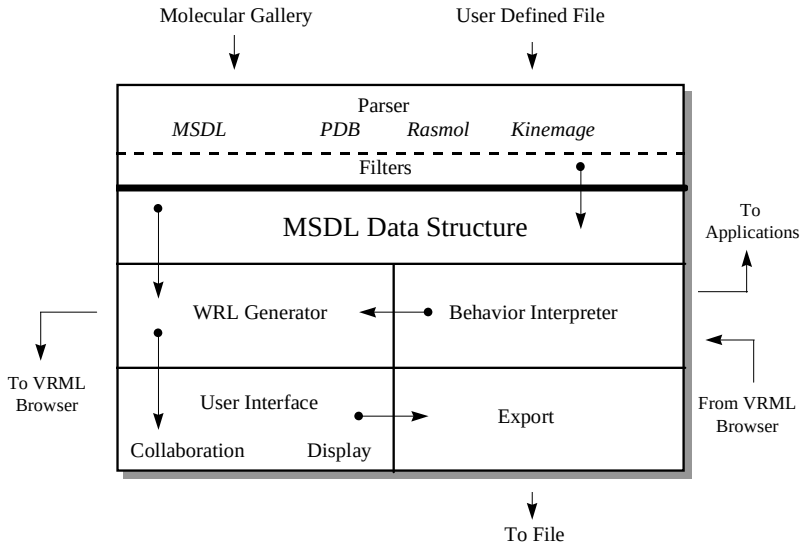

Figure 2: Java Console Components.

The major components of the Java console are shown in each box and data flow is shown by the arrows. Each component is introduced, moving from top to bottom.

- **Parser** - A variety of input formats, including MSDL itself, will be parsed to create the molecular scene. A parser for MSDL, written in Objective C[11], is available and parsers for others graphics formats are under development. A detailed PDB parser has previously been written in C++[10] and is being converted to Java.

Molecular Depiction	Annotation
Atoms and bonds: ball and stick models, space filling or CPK models, hydrogen bonding, disulfide bonds *Fields*: electrostatic potential, electron density *Surfaces*: solvent accessible surface, spherical harmonic surfaces *Secondary structure*: tubes, ribbons, arrows, cylinders, sheets *Tertiary structure*: domain boundaries	Coloring by: property, geometry, sequence similarity, structural similarity, function Numbering: atoms, residues, chains Text: linked and unlinked to molecule
Rendering	**Interaction**
shadows, reflections, transparency, materials, texture mapping, lighting, background	rotation, translation, zoom, slicing independent interaction with different molecular components

Table 1: Display and Interaction Features Controlled through the Java Console.

- **Filters** - Once parsed, the appropriate formats are converted to MSDL. PDB to mmCIF filters exist[3,5] and are being extended to support MSDL, and filters for other formats are being written.
- **MSDL Data Structure** - An efficient data structure for STAR[11] has been developed and is being converted to Java for use by MICE.
- **WRL Generator -** This requires a generalization of our prototype using the full range of VRML primitives to fully describe the complex scenes represented by MSDL. This is under development.
- **Behavior Interpreter** - Behaviors are of three types: (i) those that trigger an external application; (ii) those that update the VRML view *without* downloading a new WRL and (iii) those that generate a new WRL. Types (i) and (ii) have been developed and development of type (iii) is underway.
- **User Interface - Display** – This is being designed to emulate the PDBtool[4] interface. That is, to have interface components that mimic molecular structure. In PDBtool this meant picking with the mouse, chain identifiers, heterogens, and regions of secondary structure. Here it implies, in addition, details of the molecular scene - coloring, spline representation, sphere and line sizes and so on. This has yet to be written, however, the type of display features to be controlled through the Java console are given in Table 1.

- **User Interface - Collaboration** - Establish a connection for a collaborative session and display what other collaborators are connected, whether it is a public (anyone can join) or private collaboration, indicate who has control of the scene, and who is awaiting control of the scene. This is currently under development.
- **Export** – Exports a molecular scene in a variety of common formats, in other words, reverses the import process. This includes exporting a molecular scene in MSDL for possible inclusion in the molecular scene gallery. This is under development.

2.2.3 Molecular Scene Gallery

The long-term goal is to have the scientists who best understand structure/function relationships construct the images to be present in the gallery. As described already, these informative images will then be available as starting points for individual or collaborative exploration. This is a future goal of MICE and not discussed further.

3 Results

Initial experiments with MICE have been locally via the SDSC intranet and via standard Internet connections between SDSC and the California Institute of Technology. These experiments indicate the feasibility of this technology using VRML and the VRBS protocol.

4 Conclusion

MICE supports the advanced scientific visualization of biological macromolecules, either in a shared collaborative environment or for a single user. MICE applies VRML as a proven technology for scientific visualization. Intuitively, collaborative environments would seem to be a logical step in Internet evolution. All the software components exist to develop a collaborative system. The problem is quite the opposite. Components are emerging and disappearing so quickly as to make the choice of software difficult. Assuming a collaborative system built from persistent off-the-shelf components, will that system be useful in scientific visualization for research and distance learning? Attempting to answer this question is part of a future evaluative phase of MICE.

Acknowledgments

This work is supported by NSF grant ASC 8902825 and NIH P41 RR08605.

References

1. A. Ames, D. Nadeau, and J. Moreland, *VRML v2.0 Sourcebook*, (Wiley, 1996).
2. L. Beca, G. Cheng, G.C. Fox, T. Jurga, K. Olszewski, M. Podgorny, P. Sokolowski, T. Stachowiak, and K. Walczak (1996) *http://trurl.npac.syr.edu/tango/papers/tangowp.html.*
3. H.J. Bernstein, F.C. Bernstein and P.E. Bourne, *J. App. Cryst*., Accepted, (1997).
4. J. Biggs, C. Pu, A. Groeninger and P.E. Bourne, *J. App. Cryst*., **29**, 484 (1996). *http://www.cse.ogi.edu/DISC/PDBTool*
5. J. Biggs, C. Pu, and P.E. *Bourne Fifth International Conference on Intelligent Systems for Molecular Biololgy* Ed. T. Gaasterland et al., 52 (AAAI Press, 1997).
6. P.E. Bourne H.M. Berman, B. McMahon, K. Watenpaugh, J. Westbrook, and P.M.D. Fitzgerald, *Methods in Enzymology* **277**, 571 (1997).
7. P.E.Bourne (Ed.), *Proceedings of The Macromolecular CIFtools Workshop,* Tarrytown NY (1993*).*
8. S.E. Brenner and T.J.P. Hubbard, *ISMB* **3**, 66 (1995).
9. R. Casher, H.S. Rzepa and S. Green, *J. Mol. Graphics*, **12**, 226, (1994). (*http://www.ch.ic.ac.uk/jmg/CRG.html*).
10. W. Chang, I.N. Shindyalov, C. Pu, and P.E. Bourne, *CABIOS* **10**, 575 (1994).
11. W. Chang and P.E. Bourne, *J.App. Cryst.* Submitted, (1997). *http://www.sdsc.edu/pb/cif/OOSTAR.html*
12. A. Cook and S.R. Hall, *J. Chem Inf. Compt. Sci*. **31**, 326, (1992).
13. G.S. Couch, E.F. Pettersen, C.C. Huang, and T.E. Ferrin, *J. Mol. Graphics* **13**, 153, (1995).
14. W.N. Grundy, T.L. Bailey and C.P. Elkan, *CABIOS* **13**, 397, (1996) *http://www.sdsc.edu/MEME/meme.2.0/website/*.
15. S.R.Hall, F.H.Allan, and I.D.Brown, *Acta Cryst*. **A47**, 655 (1991).
16. Intel Corp., Proshare, (1996) *http://www.intel.com/pcoems/psvideo/potsover.htm.*
17. Molecular Simulations Inc. (1996), *http://www.msi.com.*
18. D.R. Nadeau and J.L. Moreland, *Proceedings of VRML95*, (ACM Press, 1996a). *http://www.sdsc.edu/EnablingTech/Visualization/Behaviors/vrbs_proto.html.*
19. J.L. Moreland and D.R. Nadeau, *Proceedings of VRML95*, (ACM Press, 1996b). *http://www.sdsc.edu/EnablingTech/Visualization/Behaviors/vrbs_pbs.html.*
20. D. Nadeau, C. Michaels, J. Moreland, D. Zlotin (1995), *http://www.sdsc.edu/projects/vrml/tools/webview/help/.*
21. National Center for Supercomputer Applications (NCSA, 1997) *http://www.ncsa.uiuc.edu/SDG/Software/Habanero/.*
22. D. Richardson and J. Richardson, *Protein Science*, **1**, 3 (1992).
23. D. Richardson, *TIBS*, **19**, 135 (1994).
24. R. Sayle, Rasmol (1995)
25. I.N. Shindyalov, J. Cooper, W. Chang and P.E. Bourne, *Proceedings of the 28th Annual Hawaii International Conference on System Sciences*, 207 (IEEE 1995).

26. Silicon Graphics Inc., InPerson Conferencing Software, (1996) *http://www.sgi.com/Products/software/InPerson/ipintro.html.*
27. C. Smith, M. Gribskov, I.N. Shindyalov, S.S Taylor, L. Ten Eyck, S. Veretnik, P.E. Bourne TIBS, Accepted (1997). *http://www.sdsc.edu/kinases*.