Interactive program for visualization and modelling of proteins, nucleic acids and small molecules

H E Dayringer*, A Tramontano†, S R Sprang† and R J Fletterick†

*Monsanto Company, 700 Chesterfield Village Parkway, Chesterfield, MO 63198, USA
†Department of Biochemistry/Biophysics, University of California San Francisco, San Francisco, CA 94143, USA

The graphics package Insight for the DEC VAX and Evans and Sutherland PS300, created as part of a joint university–industry research project, provides a broad set of capabilities which allow the user to display molecular models in stick figure and surface representation. The Insight program allows the user to model and manipulate proteins, nucleic acids and small molecules. The software accepts coordinate input from several possible sources and provides both a command and menu interface for manipulation of the graphics objects. The command language and program structure make it easy for the biochemist or molecular biologist to use.

Keywords: proteins, nucleic acids, small molecules, vizualization, modelling

The investigation of structure–function relationships in proteins and nucleic acids has increased the demand for sophisticated graphics systems for their visualization and manipulation[1]. The need for versatile and flexible software capable of display and manipulation of molecules ranging in size from a few to thousands of atoms was recognized as one common to both academic and industrial researchers as each pursue their interests in complex biological systems. Several excellent programs exist[1-3] or are under development. However, each has limitations such as number of objects displayed, generality, speed or ease of use. For this reason a joint development of software capable of providing the necessary support for display and manipulation of macromolecules was undertaken. The Insight software is designed to operate with an Evans and Sutherland PS300 calligraphic display device linked to a VAX minicomputer or similar host. The results of this work are described here.

OBJECTIVES

One major goal in creating a new software package was to make it as easy as possible for the novice and expert the user may choose from both command and menu input streams which are simultaneously active and provide overlapping capabilities. The command input stream allows interaction with the program via the keyboard connected to the host. This interface provides a great deal of flexibility and allows a full range of options (see Table 1) but requires typing. The menu input stream allows selection of a command or reference to an object simply by pointing to the command or object with a cursor controlled via a data pad (see Colour Plates 1 and 2). The menu input stream is very simple to learn to use, requires very little typing and is completely visual. Experience has been that both frequent and casual users adapt to the system very quickly. The casual user learns to use the menu very easily. The advanced user gets considerable power and efficiency from using the command stream as the primary input.

The authors wanted to avoid the limitations inherent in current systems that restrict the user. Thus, except for the limitations enforced by the computer hardware, there are no limits on the number of objects defined or displayed, the size of objects (number of atoms, residues or vectors), or the colouring of the objects or their surfaces (see Colour Plate 3). The C language[4] was chosen to achieve thse objectives because of its ability to manage memory dynamically.

Finally, the system was required to be extensible. Addition of new functions should be fairly simple for example; external functions such as energy minimization are anticipated and should be easily added. To this end, the software is modular and takes advantage of features of both the C language and VAX/VMS to provide simple methods for integration of functions both internal and external to the program.

CAPABILITY

The capabilities of the program can be divided into several broad classes, each of which are represented in the system by several commands or menu items. Each such class will be discussed along with representative examples. A complete list of the commands is shown

Table 1. List of commands available to the user

Command	Description
ABBREVIATE	Define new command in terms of existing commands
ADD	Add new object to global association
ALIAS	Add new alias name for object
ANGLE	Measure planar angle for three atoms
ASSOCIATE	Create new global association
AXES	Define axis system for object
BOND	Create or destroy bonds
CELL	Define unit cell parameters
CENTER	Define rotation center of object
CHARACTER COLOR	Change label colour
CHARACTER SIZE	Change label size
COLOR	Change colour of atoms, surface or object
CONNECT BUTTON	Connect button to object
CONNECT DIALS	Connect dials to object
COPY	Create a copy of an object
DELETE	Delete an object
DIHEDRAL	Measure dihedral angle for four atoms
DISPLAY	Control display of object, molecular surface or axes
DISTANCE	Measure distance between two atoms
EXIT	Leave program
EXPLAIN	Detailed explanation of last error
GET	Input new object
HBOND	Display hydrogen bonds
HELP	Get help on commands and options
IGNORE	Display control for association components
LABEL	Label atoms, residues and objects, user labels
LINK	Add and remove bonding distances in table
LIST	Summarize status of defined objects
MERGE	Combine or split molecular objects
MONITOR	Setup dynamic distance monitor
MOVE	Move object by specified amount
NEIGHBOR	Display neighbours of an atom
PUT	Output Cartesian coordinates to external ASCII file
RENAME	Change alias name for object
REPLACE	Change a residue to a new type
RESTORE	Get back objects from a save file
ROCK	Rock the object about the vertical screen axis
ROTATE	Rotate the object by specified amount
SAVE	Store complete object definition in an external binary file
SCALE	Change scale to specified value
SET	Change CSM and depth status of object
SOURCE (@)	Execute external command file
STEREO	Put displayed objects into stereo mode
SUPERIMPOSE	Superimpose two proteins
SURFACE	Generate a surface for the molecule
TORSION	Change torsional angle
VMS	Create VMS subprocess
WAIT	Wait specified time interval

The Insight program can display and manipulate both molecular and nonmolecular objects. A molecular object is one which contains atoms which are divided into residues. Each atom and residue has a name, and can be referenced in commands or selected from the screen using the data tablet. Nonmolecular objects are those which do not have atoms. A user defined object is one example of a nonmolecular because it has only vectors or dots. An association of objects (see below) is also a nonmolecular object because it is composed of a collection of other objects. Nonmolecular objects can only be referred to in their entirety. Individual lines or dots cannot be referenced.

When commands can be applied to both molecular

simply as objects. Operations applicable only to molecular objects are noted. Unless specifically noted, all functions can be executed either from the menu or command interface.

The program places no limit on the number of objects which can be defined and displayed simultaneously. The PS300 limit on stored and displayed vectors is generally reached at about 20 000 atoms for 1Mbyte of mass memory.

Store and retrieve operations

Objects for display can be input (GET[+])* from the Brookhaven Protein Data Bank[5], Cambridge Crystal Data Base[6], Frodo output[3], an amino-acid definition file or user defined vector lists. The GET command requires specification of the appropriate code name for the molecular data base type and the coordinate file name. For example, the GET PDB command specifies the four character PDB entry name. A keyword can be appended to the command line for inclusion of HETATM cards. The program does any necessary coordinate conversions and scaling, and computes connectivity. New objects can also be created by making a copy (COPY) of an existing object. Any object can be saved (SAVE) in a disc file in the program internal format for later retrieval (RESTORE). After user modification with the Insight program (see below), molecular objects can be output (PUT[+]) in either PDB format or in a format suitable for use with Connolly's analytical molecular surface programs[7]. Objects can also be flushed from the system (DELETE).

Object manipulation

These operations are divided into three categories. The first category refers to commands which manipulate the whole object. These include translation (MOVE), rotation (ROTATE) and scaling (SCALE) operations which can be done either by command, menu or from the PS300 dials. When an object is connected to the PS300 dials (CONNECT DIALS) the user has control of x, y and z rotations and translation and the scale of the object. The eighth dial is connected to the viewing window position and thickness. The user can optionally specify the centre (CENTER) and axis system (AXES) for rotation and translation; the default centre of rotation is the centre of mass. The axis system about which rotation and translation are carried out can be either the external world system or an object-based system whose default is the moment of inertia axes[8]. This category of object manipulation also includes a command to visually superimpose two proteins (SUPERIMPOSE[+]).

The second category of object manipulation applies only to molecular objects and allows internal changes in torsional angles (TORSION), addition and removal of bonds (BOND), changing the type of a residue (REPLACE), merging (MERGE) molecules and separation of a molecule into non-bonded molecular objects. Torsional changes are made by selecting the atoms involved in the rotation, and then using the PS300 dials to adjust the angle.

*The parentheses indicate a command which implements this feature

The third category of object manipulation is the creation of object associations (ASSOCIATE). An association is a loose coupling of objects which allows them to be treated as if they were a single object while, at the same time, retaining the individual objects as separate entities. ASSOCIATE applies to all objects including those that are nonmolecular. This is in contrast to MERGE which works only for molecular objects and creates a single object in which the identity of the original objects is lost. By default, control of the display status of the individual objects of which the association is composed is an attribute of the association. A command (IGNORE) is provided to allow control to revert to the individual objects. This command is used when one wants to turn off the display of a single object which is part of a global association.

When an association of objects is created, a common centre of mass and axis system is calculated. The association is treated as a single object and can be rotated, translated and scaled with the effect of these operations applied simultaneously to all objects of which the association is composed. This function of the INSIGHT program is particularly useful in visually aligning molecules. Associations can be nested such that a member of an association can itself be an association.

Parameter measurement

Spatial distances (DISTANCE), planar angles (ANGLE) and dihedral angles (DIHEDRAL) for atoms in the same or different molecular objects may be determined using the program. All of the intramolecular spatial distances for the neighbours of any atom (NEIGHBORS) within a specified distance range may be displayed, and up to 100 pairs of distances, inter- or intramolecular, may be dynamically monitored (MONITOR). The latter facility is a real-time display of the displayed spatial distances between the atoms and is very useful for monitoring docking experiments or torsional operations.

Visual attributes

An extensive set of commands are available for manipulation of visual attributes. Display of objects, residues, atoms, axis systems and the surfaces of atoms can be selectively enabled and disabled (DISPLAY). Objects can be coloured (COLOR) by specification of both a hue and a saturation value. Molecular object colouring is allowed at the residue or atom level. A special colour (COLOR PROPERTY$^+$) command allows properties of residues to be represented using colour. This allows the program to display, for example, the SHAPELY colour scheme for a protein where each of the side chains is distinctly coloured by physical property[9]. Other visual attributes include surfacing of atoms (SURFACE) with either van der Waals or Connolly[10] surfaces, and colouring these surfaces; labelling (LABEL) of atoms, residue and objects; user defined labels; changing the colour (CHARACTER COLOR) and size (CHARACTER SIZE) of the labels; rocking (ROCK) the object; and displaying alternate left/right eye images (STEREO) for use with the Terabit stereo viewer.

Parameter setting

Parameter setting options are used to setup special

Table 2. Examples of the name syntax for objects

Name string	Refers to:
ENZ	Total object (includes all atoms and residues)
ASSOC.ENZ	The ENZ components of an association
ENZ:A137	Specific residue of object
ENZ:A137–A140	Ranges of residues in an object
ENZ:A137–A170@3	Every third residue in a range
ENZ:LYS	All lysines of the object
ENZ:A137:CA	The α-carbon of a specific residue
ENZ:A137:CA,N,C,O	The backbone of a specific residue
ENZ:A137:*	All atoms of the specified residue
ENZ:A137:C*	All carbons of the residue
ENZ:*:CA	All α-carbons of the object
ENZ:B*:CB–*	All side chains of the B chain
::* or *::	All atoms of all residues of all molecules
*	All objects, regardless of type

include commands to rename (RENAME) an object or to assign additional alias names (ALIAS), specify unit cell sizes (CELL$^+$) for use in symmetry replication (SYMMETRY) and a command to change the maximum distance between atoms considered to be bonding (LINK$^+$). In addition, the PS300 colour screen mode (CSM) and depth-queueing functions can be disabled (SET$^+$) for any object and the viewing window position and thickness (z axis) can be controlled using a PS300 dial.

Command environment

A set of commands are available to allow control of the program itself. A command abbreviation feature (ABBREVIATE$^+$) allows one to define new commands or keywords in terms of existing commands. Predefined command files can be executed (SOURCE or @$^+$). Commands can be passed to the operating system (VMS$^+$) without leaving the program. A command to pause (WAIT$^+$) for some number of seconds is implemented for use with the SOURCE command files to aid in making movies.

Syntax for object specification

The user specifies, to the command interface, a name of the form:

Alias:Residue:Atom

The alias name is the name assigned to the object when it was created or through use of the ALIAS or RENAME commands. In addition to the specification of an individual object, components of an association may be referenced by specifications of the form:

Assoc.Entity

The residue and atom portions of the name are valid only for molecular objects and may or may not be required depending on the command given. Examples of name syntax are shown in Table 2. The residue specification is of the form

$$residue = type \ @\#$$
$$or = ID@\#$$
$$or = ID_1 - ID_n@\#$$

where 'type' is a one or three character abbreviation for an amino acid type (for example K or Lys) and 'ID' is an identifier, usually a residue number, but in general an alphanumeric (for example in the PDB format it is of the form 'chain# alt' as in B131A). Position specifications may be separated by commas to give single positions, or by dashes to give a range of positions. If post-fixed with @# (for example @3), every third residue is referenced.

Specification of atoms is similar to specifying residues, allowing single atoms, ranges of atoms and every nth atom to be specified. In the alias, residue and atom specification, a question mark (?) can be used to match any single character and a star (*) can be used to match any number of characters. This wild card capability is allowed, as appropriate, in any object name specification.

PROGRAM STRUCTURE

To meet the objectives of ease of use and extensibility it was necessary to design an efficient internal structure for the program. Several different systems for storing and manipulating coordinates have been used[11] in previous systems. The authors chose to store the information in a three level network database style[12] in memory using a forward linked list and queue list head implementation.

The information maintained at each level of the three levels of the memory based network data base is shown in Table 3. The per object information (level one) includes the information required to address the object in the PS300 and parameters which determine the orientation, position and size of the object on the screen. The flag word contains status information for display control and the type of the object.

Each object is assigned a unique alias name by the user when initially defined (level 2). This name can subsequently be changed or additional unique names defined as required by the user.

The display control list contains information about which atoms are displayed, and the colour in which they are displayed. Each control list consists of a header containing colour, number of atoms and a sequence number. The remainder of the list entry is a bitmap of the atoms referenced by this list. The bitmap is compact and easy to reference in the C language and vector lists for the PS300 can be constructed very quickly based on this information.

Table 3. Information stored in levels of the hierarchy (per object)

Level 1	Level 2	Level 3
PS300 name	Alias name list	Residue list
Flags	Label list	Atom list
Point to level 2 data	Display control	
Centre of mass	Surface points list	
Current rotation*	Pointer to level 3 data	
Current translation*		
Current scale*		
Axis system		
Unit cell parameters		

Surface information is held in two lists. A bitmap controls which atoms of the surface are displayed. A second structure contains pointers to coordinates for all surface points which are to be displayed in the same colour. Using this representation it is possible to change the colour or display status of an atom or its surface simply by setting or clearing bits or changing pointers.

The residue list (at level 3) contains the residue name, type, first and last sequence number of atoms in the residue and the minimum and maximum x, y and z coordinates of atoms in the residues which are used in computing connectivity. The atom list contains the atom name, x, y and z cartesian coordinates of the atom, a pointer to the residue, flags, sequence number and a table of the sequence numbers of the directly-bonded atoms.

RESULTS AND DISCUSSION

The objective in creating this software was to make a flexible, efficient, extensible program capable of aiding in the solution of complex problems in macromolecular modelling. At the same time, it was required to be convenient for use by both casual and frequent users. Through careful design of the program and data structures, the program is efficient. It requires a minimum of memory on both the VAX and PS300 and grows only to the size required to hold the objects defined. There are no fixed limits on the size of molecules. (Connections are stored in 16 bits which imposes a limit of 65535 atoms per object, but this is easily changed).

Ease of use is achieved by implementing both a menu and command interface to the program. Our experience indicates that it takes only a few hours of hands-on experience with the program before one can do esssentially anything of which the program is capable. A complete online help facility and extensive user guide are also available to aid when problems arise.

The capabilities of the Insight program are substantial. What makes this program different from other existing software is the degree of automation of useful functions and the ability to integrate external functions and subprocesses into the program. This can be illustrated by examining how the program finds atomic bonds and builds surfaces.

The speed with which bonding or connectivity can be determined is, in general, of order n^2 (where n is the number of atoms), since each atom must be compared with all other atoms except itself. For proteins or any other known polymer, this can be simplified because the internal connectivity of each portion of the structure can be encoded in a template. However, even after application of the template to protein residues the connections of the C and N atoms (and the S in Cys) are still unknown. In general one should not assume that the user has presented the residues in the proper order. Thus the problem for proteins is generally of the order (#residues)^2. These remaining connections must be found by examination, for which the program must know the maximum length of the bond. This length is a function of atom type. The Insight program allows the user to define the maximum threshold bonding distances via the LINK command. There are defaults for all of the usual protein atom pairs.

1 Clear 'connect complete' flag for each atom.
2 For each residue, compare the type and number of atoms to each template residue. If a match to the template is found, determine the atom correspondence between object and template atoms of this residue. If all atoms of the residue have a corresponding atom in the template, establish these bonds and mark as complete all but the C and N (and S of CYS) atoms.
3 If atoms remain after step 2 which are not marked complete, do step 4.
4 If more than 200 atoms remain to complete, try to use the array processor. If fewer than 200 atoms remaining or an AP is not available, use VAX Macro code.
5 AP code:
For atoms $i = 1$ to $n-1$ compute squared distance to atoms $j = i+1$ to n. If the distance squared is less than maximum value (square of largest value in LINK table), save index i and j and squared distance. On VAX: for each pair of indices saved, compare the distance found to proper entry in LINK table. Save as bonds those which are within the table distance.
6 VAX Macro:
For residues $i = 1$ to $n-1$ and residues $j = i+1$ to n determine if the distance between the residue boundaries is less than maximum (uses residues limits stored in residue block). If residues are sufficiently close to allow bonding, consider the atoms of the two residues. For each atom of residue i which is not marked complete, determine the squared distance to all incomplete atoms of residue j. For any distance which is less than the proper entry in LINK table, store the connection between these atoms in the atom data structure.

Figure 1. Algorithm for computing connectivity for molecules

molecule as well as nonstandard residues in defined polymers, and thus cannot depend on prior knowledge of the species of macromolecule or its constituent parts. Therefore, it automatically employs template or *a priori* methods as required. The algorithm is outlined in Figure 1. Both the array processor and VAX Macro code are n^2 algorithms for the atoms whose connectivity cannot be established by template matching. The VAX 11/780 CPU time required to compute connectivity for some example compounds is shown in Table 4. Use of this hybrid approach allows a general and very fast computation of connectivity.

The second example of function integration is generation and display of the molecular surface. The program can display two types of surfaces: a low or high density van der Waals surface (two or four dots per square Å) and a solvent accessible or Connolly surface[10].

The van der Waals surface is generated internally by the program from precalculated spheres of appropriate radii and density. For each atom whose surface is to be displayed, a copy is made of the sphere of proper radius and density[13]. Any point on this model sphere which lies inside the sphere of any directly bonded atom is discarded. The remaining points are stored in a memory-based list for use in generation of the picture. The data points are stored 'atom centred' (origin at zero) to facilitate torsional angles changes.

The solvent accessible surface[14] is much more computationally intensive, requiring several minutes of CPU time to surface 200 atoms. For this reason, the generation of this surface is done as a subprocess of the main program. The user indicates the atoms to be surfaced and the context within which to surface. The context includes both the atoms whose solvent accessible surface is to be illustrated as well as any other atoms that potentially contribute surrounding or intersecting surface points. Use of a context makes it possible to surface an inhibitor bound to an enzyme ignoring the presence of the enzyme atoms.

All atoms to be surfaced and the atoms of the context are written and transferred to a disc file. A subprocess is created and the commands to run the Connolly programs are passed to it. When the subprocess is complete, it causes an interruption to the owner process. The surface file, containing the coordinates of the surface points, is processed and the coordinates stored in lists identical to those used for the van der Waals surface. When fully processed, the surface is automatically added to the display. Using this integrated approach the user can obtain the surface, van der Waals or solvent accessible, without leaving the Insight program.

FUTURE DEVELOPMENT

Plans for future developments include interfacing the Insight program to external energy calculations and simulation programs, as well as integrating an energy evaluation function for real-time energy display. The authors also plan to build in a number of features to enhance the model building capability of the program including the ability to insert and delete residues.

To summarize, graphics software for the modelling of chemical compounds has been created which is completely general and whose capabilities can be easily expanded. The program has a simple and powerful user interface, based on either command or menu input, making it possible for both the novice and expert user to get the most out of the system. There are no limits on the number, or size of the objects displayed, their colours or the surfacing. Objects can be associated to create new objects which can then be treated as if they were single entities. Full advantage is taken of the features of both the C language and the VMS operating system to make the program efficient in its use of computer and graphics resources. The software is available, including the sources, from Biosym Technologies[15].

REFERENCES

1 Langridge, R et al. 'Real time color graphics in studies of molecular interactions' *Science* Vol 211

Table 4. Computation of connections

Number atoms		VAX 11/780 CPU Time (s)		
In object	After template	Without AP	With AP*	Compound
10	2	<0.01	N/A	Amino PHE
94	94	0.18	N/A	Cambridge CARHIM
327	92	0.40	N/A	PDB 1CRN
2972	748	13.87	8.06	PDB 5API
6651	1632	63.50	31.60	Phosphorylase

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

fastcase®
Smarter legal research.