

SBEVSL -- Structural Biology Extensible Visualization Scripting Language Cumulative Annual Report

Periods:

Year I: 1 August 2006 – 31 July 2007

Year II: 1 August 2007 – 31 July 2008

Report date: 7 July 2008

Grant: 1 R15 GM078077-01

PI/PD: Herbert J. Bernstein, Dowling College, Oakdale, NY
yaya **at** dowling **dot** edu

Co-PI: Paul A. Craig, Rochester Institute of Technology, Rochester, NY
pac8612 **at** rit **dot** edu

Title: SBEVSL -- Structural Biology Extensible Visualization Scripting Language

Aims

The goal of the SBEVSL project is to create a new extensible scripting language for molecular graphics, as used in structural biology, by combining the intuitive expressive power of the widely used scripting language created by Roger Sayle for RasMol with the general object-oriented extensibility of the Python scripting of PyMOL. To be successful a molecular graphics program must provide a mechanism to select atoms, residues and chains and must be able to render wireframe, CPK, ball and stick, cartoon and other standard presentations. Rather than impose a language on any program, the SBEVSL project will extract all the concepts used in the command languages of major molecular graphics programs (Fig. 1) and gather them in one master ontology, using this essential dictionary as a relational database with CIF. Defining SBEVSL in terms of the dictionary and UML will allow the expression of scripts in multiple formats so that SBEVSL can be widely used. The specific goals for the first year were work on: SBEVSL dictionary, PyMOL modifications, Jmol modifications, RasMol modifications, scripts and modifications for other programs and creation of the SBEVSL web site. In doing this work, after resolving project dependencies, we gave priority to working on the more frequently used commands and features before addressing the less frequently used ones. The original specific goals for the second year were SBEVSL dictionary review and extension; further PyMOL, Jmol and RasMol modifications; scripts and modifications for other programs; common scripts; and continued development of the web site.

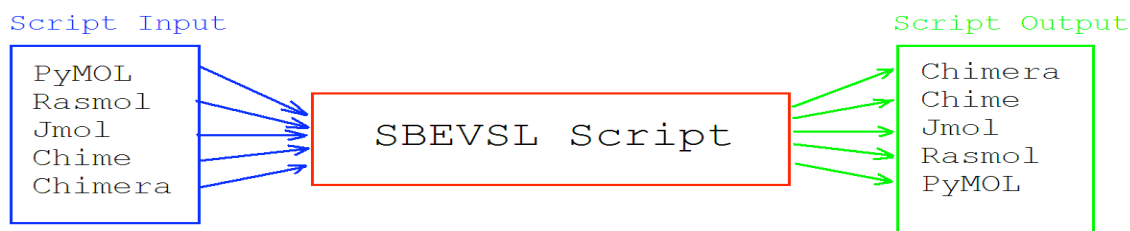


Figure 1: SBEVSL Schematic.

Progress

General Approach: We are taking a program-independent, general view of functions and features required to do molecular graphics, drawing on best practices from existing packages. Initial efforts were directed at identifying the abilities of various molecular modeling programs to determine which functions in each would have a corresponding counterpart in each of the other programs. At RIT, the molecular modeling programs PyMOL and Rasmol were the first programs

addressed in this manner. At Dowling College, the programs RasMol, PyMOL, Dino and ccp4mg were considered. In the first year, the major focus in coding for the two groups was on changes to PyMOL at RIT and changes to RasMol at Dowling College. In the second year, Dowling started to look at PyMol issues and RIT at RasMol issues, and both groups have considered Jmol. As functions and features are identified, the SBEVSL dictionary is being extended. Functions supported only by RasMol or only by PyMOL were identified and, when possible, attempts were made to accommodate these functions in the programs from which they were absent, e.g. by adding the RasMol-style hydrogen-bonding scheme and sticky selection to PyMOL, and by adding PyMOL-style map support and transient selections to RasMol. The groups hold a joint weekly e-meeting using an RIT server for Acrobat Connect (formerly Macromedia Breeze) and share code and documents via a Dowling College server for GFORGE (<http://blondie.dowling.edu>), and now via the widely used open source community server sourceforge (<http://sourceforge.net/projects/sbevsl>). The two groups meet once per year, near the start of the academic year, to discuss plans and assignments for the coming year.

SBEVSL Dictionary: Work on the dictionary is well underway. The main work in the first year was to create the basic object-oriented structure of the dictionary. The main work in the second year was to re-orient the dictionary to align with the newly proposed IUCr Dictionary Definition Language with methods. The lead in the work on the dictionary in the first year was in the Dowling group. By the end of the second year, wider participation was achieved.

In the first year, a basic object-oriented structure was proposed as an extension to CIF; the necessary scene object definitions were proposed in an SBEVSL dictionary as an extension to the mmCIF and imgCIF dictionaries. Near the end of the first year an important opportunity arose when the International Union of Crystallography proposed the adoption of a new enhanced methods-rich Dictionary Definition Language, DDLm (see <http://www.iucr.org/iucr-top/cif/ddlm/index.html>). While SBEVSL demands stronger object-oriented capabilities than are needed for the IUCr DDLm project, both projects benefit by working towards compatibility and interoperability. In addition to the SBEVSL funding from the NIH, the PI also has funding from the International Union for Crystallography for work on DDLm, so these goals should be achievable.

Because maps are of more general interest than just for molecular graphics, the necessary basic map definitions needed for SBEVSL have been integrated into the imgCIF dictionary. The preliminary versions of the dREL and DDLm methods extensions to CIF by Syd Hall, Nick Spadaccini and John Westbrook were released in April 2007 (see <http://www.iucr.org/iucr-top/cif/ddlm/>) and work began on integration of the SBEVSL object-oriented extensions to CIF with the dREL/DDLm methods extensions. The need to integrate with dREL/DDLm have caused some delays in the formal release of the SBEVSL dictionary, but those issues should be resolved during discussions with the Committee on the Maintenance of the CIF Standard (COMCIFS) at the 2008 IUCr meeting in August 2008 at Osaka Japan, and strong compatibility between DDLm and SBEVSL should speed ultimate adoption of both.

In the meantime we have reordered priorities and are focusing on the external presentation of the SBEVSL scripting language and have posted a draft external presentation document to the web. The main design feature of the external presentation is a menu-like model similar to the one used in RasMol's scripting language in which scene objects have "sticky" selection status and parameters have sticky defaults. This has worked well and we have provided support for this approach in PyMOL. This approach allows subsequent commands to be applied with a minimum of arguments, much like clicking on a menu item. We have also added support for PyMol-style temporary selections in SBEVSL and tested the approach in RasMol. The model for the external presentation of SBEVSL is:

```
{(<selection>){.}}<command> {<parameters>}
```

as in

```
cartoon
```

or

(14-25, 39).color CPK

to apply a cartoon rendering to the current selection, or to apply CPK coloring to the temporary selection of residues 14 through 25 and 39.

The most recent effort on the dictionary has been on support for animations and movie making. The approach we are taking is to provide simple, intuitive record, play, defer and execute commands as an alternative to existing cryptic commands that have to be rote memorized. (See below).

PyMOL Modifications: The SBEVSL PyMOL team has made excellent progress on modifications to PyMOL. The lead on this work has been in the RIT group, which began their PyMOL work under other funding prior to the start of this project. The Dowling group is now also contributing to the PyMOL code. The effort on PyMOL has had two directions: working with motif identification as a test case, and using plugins to extend PyMOL to support the common SBEVSL scripting language. The work on using RasMol commands in a PyMOL plugin began at RIT in another project prior to SBEVSL. In the first year of the SBEVSL project, the work on RasMol and then SBEVSL command support continued and development of the motif test-bed (ProMOL) with a simplified menu interface was done. In the second year, ProMOL was improved. A new plug-in for PyMOL, called ConSCRIPT, was developed which provided a first pass at the SBEVSL support in PyMOL.

In order to support a common scripting language, two major extensions to PyMOL were necessary: support for RasMol-style hydrogen bond calculations, and support for “sticky” selections. The hydrogen bond calculation was done in the first year. The support for “sticky” selections was completed in the second year, and the SBEVSL script support was also extended to PyMOL command-line support for SBEVSL commands.

PyMOL and RasMol have differing methods of identifying hydrogen bond schemes within PDB files. To provide correct transference of hydrogen bonding information from RasMol's calculations to PyMOL's viewing screen it was necessary to develop a script for the calculation of hydrogen bonds in PyMOL comparable to that of RasMol. This was done by identifying donor and acceptor atoms, and by identifying length and bonding restrictions (Figs. 2 and 3).

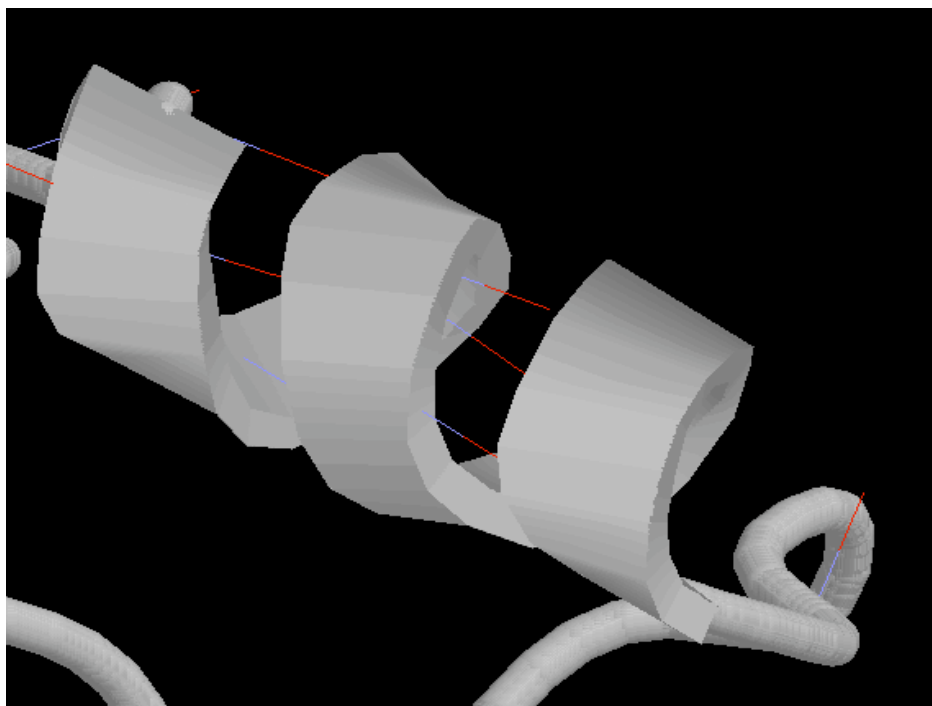


Figure 2: RasMol generated hydrogen bonds.

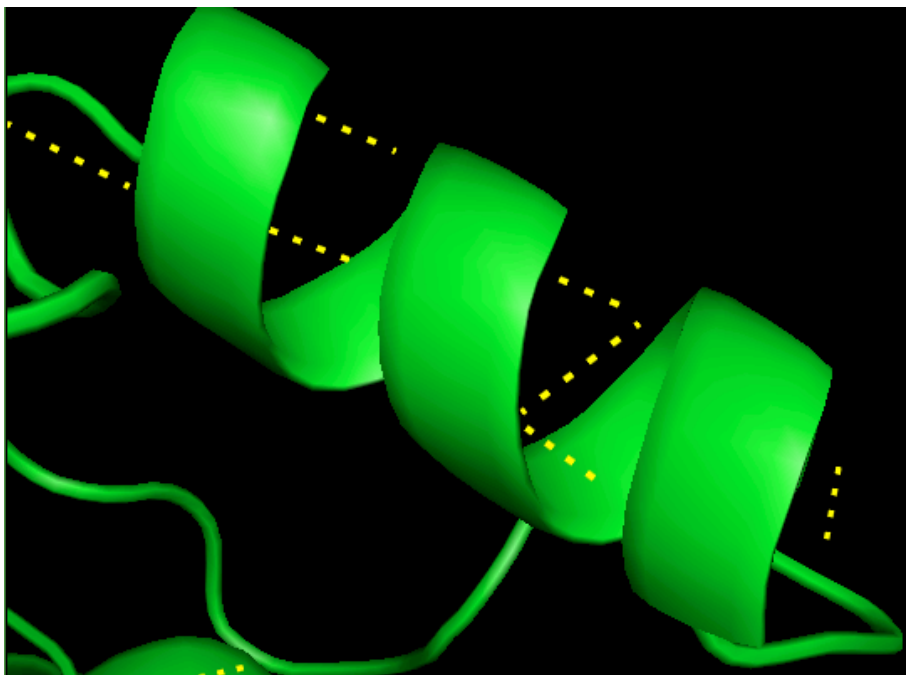


Figure 3: Hydrogen bond scheme in PyMOL, programmed to approximate Rasmol's hydrogen bonding scheme

Further efforts entailed writing modules for PyMOL to set up a framework to enable PyMOL to handle the input and output of the SBEVSL language. This was accomplished by writing two separate plugin modules for PyMOL. One of these modules (Fig. 4) was constructed so that once a SBEVSL script is output from another molecular modeling program it can be directly read by PyMOL, and, upon recognition of the SBESVL commands, PyMOL will execute the corresponding commands defined by the plugin. The other module was structured to convert PyMOL scripts to SBEVSL scripts on a line-by-line basis (Fig. 5). This new script file will eventually be readable by other molecular modeling programs, once support for the SBEVSL language has been added to them.

```
#Define the translate-out Function
def translate_out(Event):
import tkinter as tk
import tkinterFileDialog
#Open the script write a blank line
Q = tkinterFileDialog.askopenfilename(initialdir='./startup')
f = open(Q, 'a')
f.write('\n')
f.close()
f = open(Q, 'r')
Y = tkinterFileDialog.askopenfilename(initialdir='./startup')
r = open(Y, 'w')
#Make a loop
running = True
while running:
#Read each line and see if line contains these variables
p = f.readline()
if p == "cmd.zoom\n":
r.write('SBEVSL zoom\n')
```

Figure 4: Code written to facilitate translation from PyMOL script files and into the SBEVSL language.

Since these module frameworks preceded full construction of the SBEVSL language it was necessary to test the abilities of each module. This was done by enabling one of the module frameworks to handle direct Rasmol to PyMOL translations. The module is run by PyMOL as a plugin, reading a Rasmol script file line by line. If the module finds a supported Rasmol command then it then executes the PyMOL command paired with it in the program code. In this way it is possible to set up a specific view in Rasmol, write out a script file, and replicate the same view in PyMOL for further manipulation. A very similar approach will be used for other molecular visualization applications once the SBEVSL language is complete and ready for translation.

```
#Define the translate-in Function
def translate(Event):
    import tkinter as tk
    #Open the script write a blank line
    Q = tkFileDialog.askopenfilename(initialdir='./startup')
    f = open(Q, 'a')
    f.write('\n')
    f.close()
    f = open(Q, 'r')
    #Make a loop
    running = True
    while running:
    #Read each line and see if line contains these variables
        p = f.readline()
        if p == 'SBEVSL zoom':
            cmd.zoom
```

Figure 5: Code written to allow PyMOL to directly read SBEVSL scripts that originated from another molecular modeling program.

The simplified menu interface to PyMOL has been released by the RIT group in the form of the ProMOL plugin to PyMOL (<http://ase-web.rit.edu/~ez-viz/>). The ProMOL work was presented in an oral presentation by Brett Hanson at the ACA meeting in Salt Lake City in July 2007. Brett Hanson won an AIP award for undergraduate research presentations for his talk.

In the second year, ProMOL was improved and the lessons learned were reflected in a new PyMOL plugin, ConSCRIPT, that accepts SBEVSL scripts reads SBEVSL commands from the PyMOL command line as well. A fragment of the necessary Python code is shown in Figure 6, and the handling of an SBEVSL compliant RasMol script is shown in Figure 7.

The work on ConSCRIPT was presented by Scott Mottarella as a poster by at the ACA meeting in Knoxville in June 2008. The continued work on ProMOL was presented as a poster by Corey Wischmeyer at the same meeting.

Jmol Modifications: Work on Jmol modifications began in the second year. That work began with a careful review of the current Jmol commands. Inasmuch as Jmol uses the RasMol command language, the necessary changes are small. The major additions needed are support for temporary selections and support for the new RasMol-style maps which have proven particularly useful in small angle X-ray scattering work (SAXS). Work on these issues is in progress.

```
##-----View Options-----##

selectDict = {'wireframe': 'lines', 'cartoon': 'cartoon', 'dots': 'dots', 'cpk': 'spheres',
              'spacefill': 'spheres', 'trace': 'ribbon', 'ribbon': 'ribbon'}
firstword = p.split( ' ', 1 )[0].lower()

try:

    if firstword in selectDict:
        q = p + ' '
        command = q.split( ' ', 1 )[1][:-1].lower()
        if command=='false' or command=='off':
            cmd.hide( selectDict[firstword], selected)
            print firstword + ' off complete'
        elif command=='true' or command=='on' or command=='':
            cmd.show( selectDict[firstword], selected)
            print firstword + ' on complete'
        else:
            print 'That function is not supported by PyMOL'

    except:
        print 'An error has occurred with your view command.
        Please check your spelling in your command,
        the selection on which this command was executed
        and that your command is supported by PyMOL.'
```

Figure 6: Code written in ConSCRIPT to allow PyMOL to directly handle several of the commands in SBEVSL scripts that originated from RasMol.

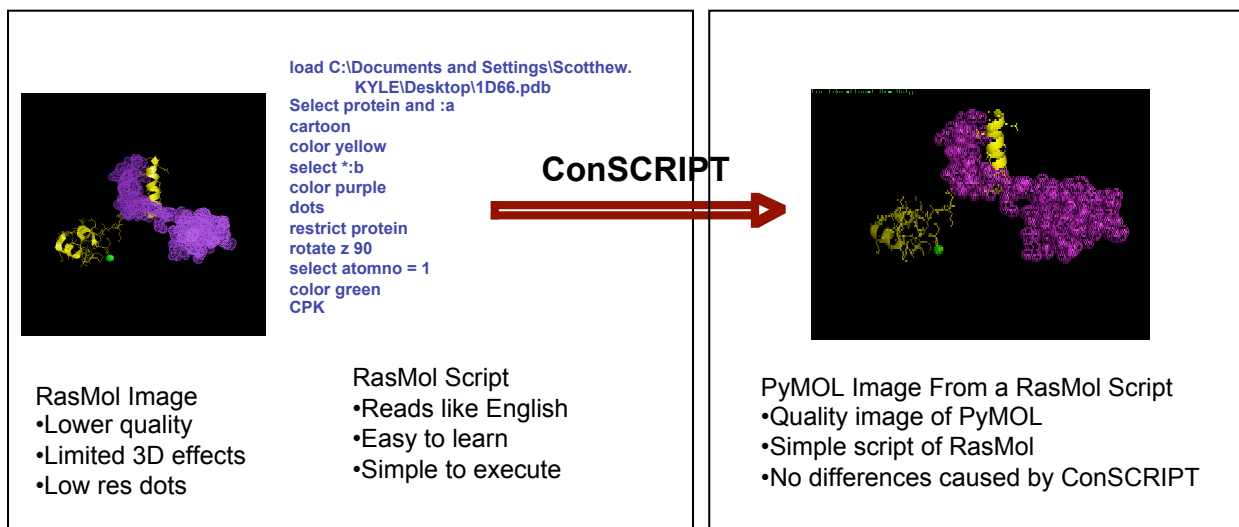


Figure 7: Example of an image from an SBEVSL-compliant RasMol Script translated by ConSCRIPT for display in PyMOL

RasMol Modifications: The major change needed to RasMol for this project was the addition of support for maps (Fig. 8). The lead in this work has been in the Dowling group and began in the first year. The first step was the addition of the necessary map support to the imgCIF dictionary and to the CBFlib API for imgCIF. Part of this work was funded by NSF and DOE. The dictionary extensions and code were released to the community for comment. The next step was the modification of RasMol to use both CBFlib and the CCP4 maplib. A preliminary version

was presented in a poster at the ACA meeting in Salt Lake City in July 2007. Discussions at the meeting on the subject indicated that the changes are desirable. However, one complication that arose shortly before that ACA meeting was a web discussion on licensing of the CCP4 maplib. We initiated discussions with CCP4 to try to resolve this issue, and we are pleased to report that by the end of the second year, CCP4 had adopted the GNU LGPL for maplib. However, in order to avoid a delay to SBEVSL while this was being resolved, we provided an LGPL'ed (Lesser GNU Public License) replacement for the portions of maplib that we required.

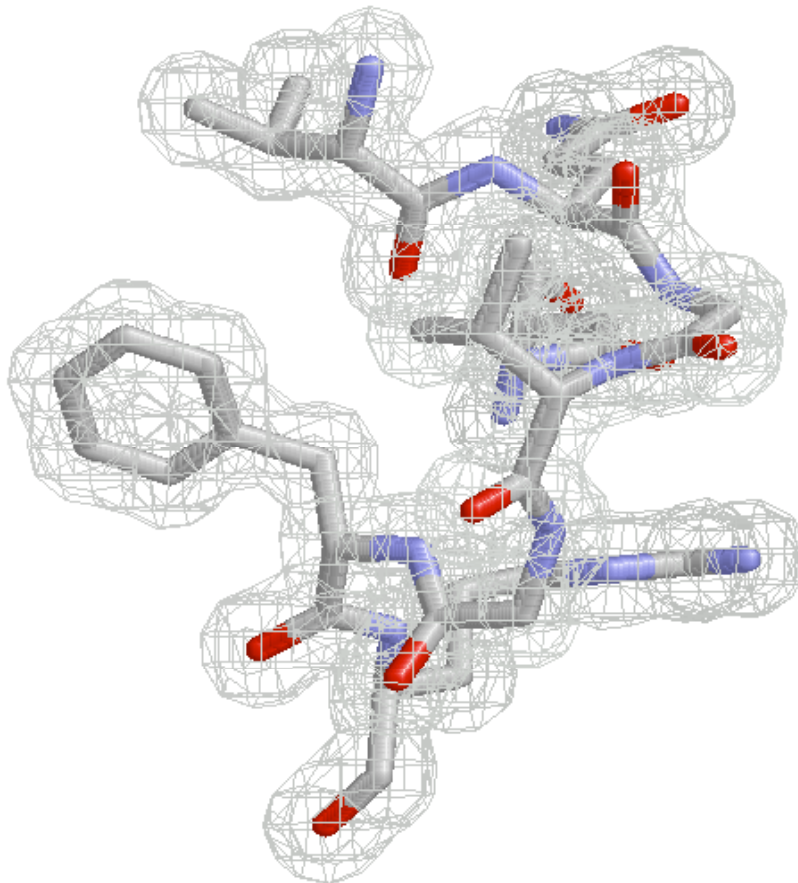


Figure 8. How RasMol Displays a Map. This is RasMol display of generated map and 1w2i.pdb, residues 14-20 and 38 from Chain A.

At a second presentation of the new map logic as a poster at the Synchrotron Radiation in Biology meeting (BSR 2007) in August 2007 at Manchester England by Nikolay Darakev, there was strong interest in this approach to maps for use in SAXS, and the defaults were adjusted appropriately to favor this use. Intensive work subsequent to that meeting resulted in a simple resolution-based command set which has been incorporated into SBEVSL, and released in RasMol (version 2.7.4.2). Since that release (as a pre-release in November 2007 and as a full stable release in March 2008) downloads of RasMol have risen to over 2000 downloads per week from approximately 800 downloads per week before this release. The proposed command set for maps was presented by the PI as a poster at the ACA meeting in Knoxville in June 2008.

The following SBEVSL command sequence generated a SAXS envelope representation (Figure 9) in RasMol 2.7.4.2 from beads created by the Situs program pdb2sax:

```
load 1w2isax.pdb          # load the coordinates
map resolution 3.         # 3 A spacing, spread of 2
map generate surface     # create the surface
set specular on          # adjust the illumination
slab 75                  # peek inside the surface
load 1w2i.pdb            # load another copy
```

```
wireframe off           # turn off the initial wireframe
cartoon on             # draw a cartoon
color structure        # color by structure
background [128,128,255] # pick a background
set stereo on         # view it in stereo
```

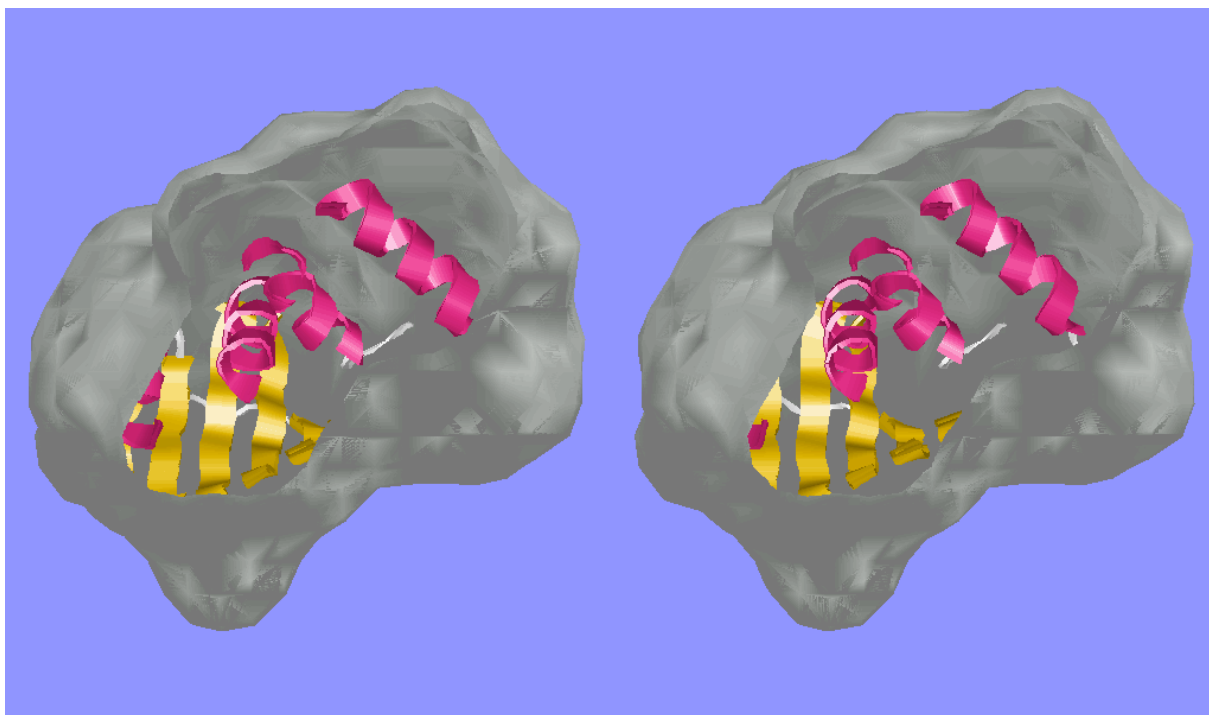


Figure 9. RasMol 2.7.4.2 image of 1w2i [Cheung et al. 2004] surrounded by 3 Angstrom resolution surface generated from beads using pdb2sax [Wriggers, Chacón 2001]

Consolidation of Animation and Movie-Making Commands: Each of the programs can be used to produce animations and movies. However, for morphing, the typical practice is to have an external program generate a sequence of molecular images as if they were NMR models, and then to have the molecular graphics program render those models in some selected order. This leads to some complexity in setting up animations and movies that combine externally generated morphing with locally selected choices of images in particular static positions (poises). PyMOL approaches this issue with separate “state” and “frame” abstractions. Jmol packages various common runs through choices of models. Jmol provides a “moveto” abstraction to generate frames in a simplified linear morph through a sequence of poises. For those who are familiar with these commands, they are very useful and are being added to SBEVSL. Though more verbose, it is conceptually simpler and allows scripts to be better self-documenting to use variables, arithmetic and loop constructs as alternatives to these somewhat cryptic command structures. In many cases, we should be able to achieve that clarity without loop constructs by allowing sets of positioning commands to be gathered together for subsequent execution as a combined group. The fundamental commands that are needed to meld the PyMOL and Jmol constructs into a movie-making context are:

Set up to record

```
record {on|off|eject} {<type>} <medium>
record {from|until} <time>
record <what> {on|off}
```


Set up to play

play {on|off|eject} {<type>} <medium>

play {from|until} <time>

Group commands to be acted on together

defer {<name>} command to defer

execute {<name>}

Where the <medium> could be a file template for, say, a sequence of SBEVSL scripts or a sequence of image frames, and <what> could be a list of commands or image states that should trigger the recording of particular images, e.g. mouse movement producing a translation of so many pixels or Angstroms or a rotation of so many degrees.

The simplified animation and movie command set is under development and a detailed proposal will be presented at the IUCr meeting in Osaka in August 2008.

Scripts and Modifications to other Programs: We have began exploring Dino and CCP4MG. The issues for Dino are similar to the ones for PyMOL, especially in dealing with a transient selection paradigm, but made more difficult because Dino is closed-source. CCP4MG is better documented and python-scriptable. Therefore CCP4MG is being pursued before Dino.

SBEVSL Web Site: The SBEVSL web site consists of multiple sites. Dowling is hosting two sites, <http://arcib.dowling.edu/sbevsl> for general public information on the project and <http://blondie.dowling.edu/projects/sbevsl> for document and code exchange and internal coordination. RIT is hosting the ProMOL site at <http://ase-web.rit.edu/~ez-viz/> and an Adobe Connect meeting site that is used for weekly e-meetings between the groups. A sourceforge sbevsl site with a wiki has been added.

Publications 1 August 2006 – 30 July 2007: In this period the project produced web pages and meeting presentations. In addition to the web pages above, this grant has been credited on the CBFlib 0.7.8 web site at http://www.bernstein-plus-sons.com/software/CBFlib_0.7.8. The meeting presentations were:

B. Hanson, C. Westin, L. Slatest, P. A. Craig, "EZ-Viz Version 2.0", Northeastern Regional Meeting of the American Chemical Society, Binghamton, NY, October, 2006.

C. Westin, B. Hanson, P. A. Craig, "Using PyMOL's Align Feature to Create a Database of Ligand Binding Site Files for the Structural Analysis of Proteins," ASBMB Annual Meeting, Washington, DC, April 28 -- May 3, 2007.

B. R. Hanson, C. Westin, P. A. Craig, "Using PyMOL's Selection Algebra for Enzyme Catalytic Site Prediction," ASBMB Annual Meeting, Washington, DC, April 28 - May 3, 2007.

B. Hanson, C. Westin, L. Slatest, P. Craig, "ProMOL, Simplification and Increased Functionality of PyMOL," presentation at session SP.01.06, American Crystallographic Association Meeting, 21-26 July 2007, Salt Lake City Utah, abstract W044.

H. J. Bernstein, I. Awuah Asiamah, G. Darakev, J. Jemilawon, P. Kamburov, "Support for Maps in RasMol," poster T174, American Crystallographic Association Meeting, 21-26 July 2007, Salt Lake City Utah, abstract W0059.

C. Westin, B. Hanson, H. J. Bernstein, I. Awuah Asiamah, D. Boycheva, G. Darakev, N. Darakev, J. Jemilawon, N. Jia, P. Kamburov and G. Todorov, P.A. Craig, S. Mottarella, "SBEVSL: Communicating Scripts Between Molecular Visualization Programs" poster T172, Crystallographic Association Meeting, 21-26 July 2007, Salt Lake City Utah, abstract E0003.

Publications 1 August 2007 – 30 July 2008: In this period the project produced web pages (see above) and meeting presentations. The meeting presentations were:

SBEVSL Cumulative Annual Report, 7 July 2008

J. Jemilawon, I. Awuah Asiamah, H. J. Bernstein, G. Darakev, N. Darakev, P. Kamburov, "Use of CBFlib for Map Support," poster presentation at poster session B, 9th International Conference on Biology and Synchrotron Radiation (BSR 2007), 13-17 August 2007, Manchester, UK.

S. E. Mottarella, C. Wischmeyer, B. Hanson, C. Westin, P. A. Craig, "Execute RasMol Scripts in PyMOL", poster American Society for Biochemistry and Molecular Biology, San Diego, CA, 5 – 9 April 2008.

S. Mottarella, B. Hanson, C. Westin, P. Craig, H. Bernstein, "A Rasmol to PyMOL Translator," poster TP161, American Crystallographic Association Meeting, 31 May – 5 June 2008, Knoxville, TN, abstract W0049

C. Wischmeyer, P. Craig, H. Bernstein, "Homology Exploration with ProMOL," poster SP169, American Crystallographic Association Meeting, 31 May – 5 June 2008, Knoxville, TN, abstract W0050.

P. A. Craig, S. E. Mottarella, C. Wischmeyer, H. J. Bernstein, I. Awuah Asiamah, D. Boycheva, G. Darakev, N. Darakev, P. Gozo, J. Jemilawon, N. Jia, P. Kamburov, G. McQuillan, D. O'Brien, G. Todorov, "Using RasMol, PyMol and Jmol with the Structural Biology Extensible Visualization Scripting Language (SBEVSL)," poster MP054, American Crystallographic Association Meeting, 31 May – 5 June 2008, Knoxville, TN, abstract W0073.

H. J. Bernstein, I. A. Asiamah, G. Darakev, N. Darakev, J. Jemilawon, N. Jia, P. Kamburov, G. McQuillan, G. Todorov. "SAXS, Lee-Richards Surfaces and a Simplified Command Set for Maps," poster MP039, American Crystallographic Association Meeting, 31 May – 5 June 2008, Knoxville, TN, abstract W0070.

Software: ProMOL and ConSCRIPT have been posted both to the CVS repository at <http://blondie.dowling.edu/projects/sbevsl> and to the SVN repository at <http://sourceforge.net/projects/sbevsl>. The SBEVSL contributions to RasMol are incorporated into the official releases of rasmol at <http://www.rasmol.org>.

The SBEVSL project website is available at <http://www.sbevsl.org>

The SBEVSL project wiki is available at <http://sbevsl.wiki.sourceforge.net/>

Undergraduate Students: The undergraduate students who have contributed to the project at Dowling College have been Isaac Awuah Asiamah, Darina Boycheva, Georgi Darakev, Nikolay Darakev, John Jemilawon, Nan Jia, Gregory McQuillan, Daniel O'Brien and Georgi Todorov. The undergraduate students contributing to the project at RIT are Brett Hanson, Scott Mottarella, Charlie Westin and Corey Wischmeyer. As of the end of the second year, Brett Hanson, Charlie Westin, John Jemilawon and Georgi Todorov have graduated. Elena Zlateva and Petko Kamburov joined the project at Dowling College in late Spring 2008.

Graduate Student: Anthony Corbett is an M.S. student in bioinformatics at RIT who joined the project in Spring 2008.

Conclusions: The project is on track and making good progress.

Glossary:

CIF	Crystallographic Information Framework
dREL	A relational expression language
DDLm	A CIF dictionary definition language with support for methods
GFORGE	An open source collaborative development environment
imgCIF	Image-supporting CIF, otherwise known as CBF , the Crystallographic Binary Format
SBEVSL	Structural Biology Extensible Visualization Scripting Language
UML	Unified Modeling Language