

## GENERATING INTERACTIVE MOLECULAR DOCUMENTARIES USING A LIBRARY OF GRAPHICAL ACTIONS

P. PULAVARTHI, R. CHIANG, R.B. ALTMAN  
*Stanford Medical Informatics, 251 Campus Drive MSOB x215,  
Stanford University, CA 94305-5479, USA  
{prasanth, chiang, rba}@smi.stanford.edu*

Paper-based publishing of scientific articles limits the types of presentations that can be used. The emergence of electronic publishing has created opportunities to increase the range of formats available for conveying scientific content. We introduce the Graphical Explanation Markup Language, GEML, implemented as an XML format for defining molecular documentaries which exploit the interactive capabilities of electronic publishing. GEML builds upon existing molecular structure definitions such as the Protein Data Bank (PDB) standard file format. GEML provides a library of gestures (or actions) commonly used for structural explanations, and is extensible. XML allows us to separate explicit statements about how to highlight a molecular structure from the implementation of these instructions. We also present GEIS (Generator of Explanatory Interactive Systems), a program that takes as input a GEML documentary definition file and produces all the files necessary for an interactive, web-based molecular documentary. To demonstrate GEML and GEIS, we constructed a documentary capturing the difficult 3D notions expressed in two selected published reports about human topoisomerase I. We have created a prototype Java application, GEMLBuilder, as an editor of GEML files.

### 1. Introduction

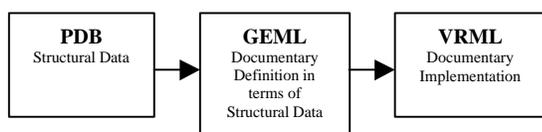
As publication of scientific literature migrates from paper to electronic format, more presentation styles and techniques are becoming available. One of the main limitations of paper-based publishing is its static nature. Pictures capture only one snapshot view of what the author intends to point out. On paper, it is not possible to rotate an image of a molecule and point out a feature on the other side. Nor is it possible to smoothly demonstrate how a reaction occurs as two molecules come together or break apart other than by showing before and after graphics. When authors use multiple views to show different orientations of a molecule, there is no cohesiveness between the various views; the separate views may seem like different molecules.

There are already many programs and viewers available which allow visualization of molecules. The PDB provides predefined views of 3D representations of the molecules in its database.<sup>1</sup> MolScript creates files in many graphical formats from a PDB file.<sup>2</sup> RasMol is a program which renders a 3D structure directly from a PDB file, and is available on multiple platforms.<sup>3</sup> MAGE is a 3D structure display program that was used as a standard by Protein Science for

parallel distribution of 3D images associated with that journal.<sup>4</sup> Without some minimal guidance, the user may have no sense of what parts of the molecule are important. Therefore, some of these programs allow special annotations and views. However, the task of creating these specially annotated models can be non-trivial, especially when the interactions become complex, so often these programs are used only to generate generic 3D images.

With the emergence of online publishing, it has become possible to incorporate animation and multimedia in the presentation of scientific material. Quinn, et. al. present an overview of the types of multimedia tools currently available and suggest ways to combine these tools to write protein documentaries.<sup>5</sup> There are some obstacles to the large scale adoption of these techniques. It is often difficult to show the important connection between the descriptions and the visual model of the molecule. In addition, each presentation is built from scratch, which can lead to inconsistent interfaces, and considerable user uncertainty.

Interactive documentaries are often created using the documentary features of 3D molecular display programs. They can also be created directly in VRML, a standard for 3D rendering on the web. Both these approaches combine the structural information about the molecule with other textual and “highlight” information. The PDB file format can be considered the lowest layer, which describes molecular structure. Languages such as VRML are another layer, which implement the details of the molecular documentary (See Figure 1). We sought to develop an intermediate layer that would describe the documentary in terms of the structural data but be independent of any implementation. XML was designed to provide such a separation between description and implementation (See <http://www.w3.org/XML/>). GEML (Graphical Explanation Markup Language) is an XML Data Type Definition (DTD) which captures many of the key actions that are required when explaining the significance of molecular structural features.



**Fig. 1.** Layers of Molecular Documentation

a PDB file containing the structure of the molecule and an XML file containing the organization of the documentary using the GEML DTD. The output is a set of HTML and VRML files that together constitute an interactive documentary which unites the textual descriptions with a 3D visual model. The documentary manipulates the structure, provides viewpoints, and uses animation to guide and focus the user through the important features of the molecule. The user also maintains the freedom to explore the molecule on his own, even as actions are

In order to parse and interpret GEML files (in the context of their associated PDB file) we have written GEIS (Generator of Explanatory Interactive Systems). To create a documentary, GEIS requires

occurring within the world. All documentaries produced by the same generator will have the same presentation style and interface, reducing user uncertainty about interface idioms across documentaries.<sup>6</sup> Our work also builds on PDB format files which, in general, do not require modification to be used with our documentary generator.

## 2. Methods

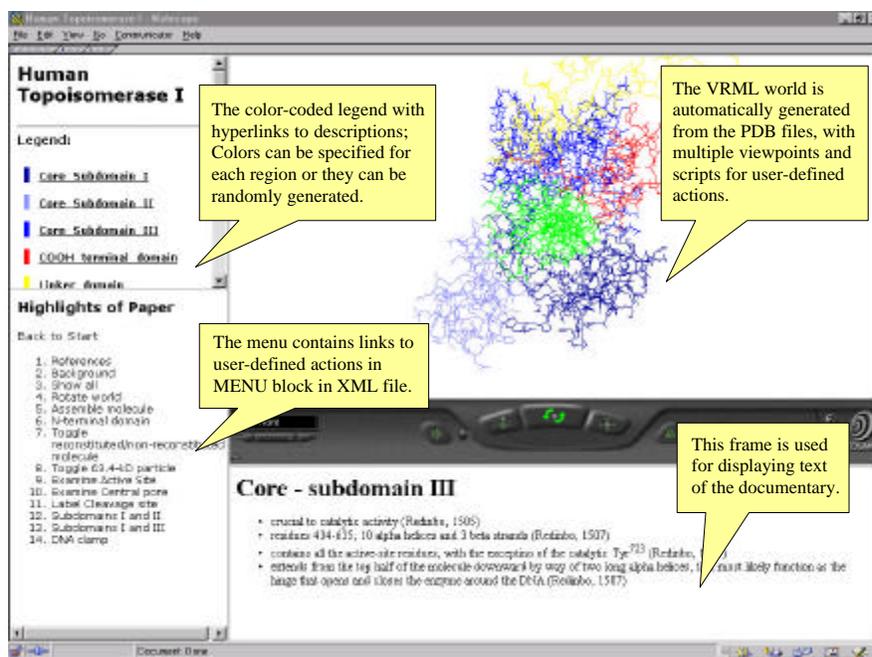
We developed GEIS with the Internet and open standards in mind. GEIS has been written entirely in Perl 5 to provide substantial platform independence and to take advantage of Perl's extensive built-in support for text processing. Our goal for the outputted documentary is to reach the largest audience possible on a wide range of platforms while still providing a common, rich experience. There is a tradeoff between targeting a powerful environment available to only a few users, and a generic environment available readily to many. We sought to find a balance by making use of those standards having implementations in the latest versions of available software. This permits our documentaries to work properly on most platforms running the latest software.

GEIS produces documentaries using standard HTML (See <http://www.w3.org/MarkUp/>), VRML97 (VRML 2.0) (See <http://www.web3d.org/Specifications/VRML97/>), JavaScript (ECMAScript) (See <http://www.ecma.ch/>), and Java. Documentaries generated by GEIS can be viewed on platforms that have a web browser supporting these features. The documentaries produced by GEIS have been tested to work on Windows 9x/NT, Macintosh, and SGI/IRIX machines. A frames-capable web browser which supports JavaScript (ECMAScript), Java and the ability for Java applets and JavaScript to interact (called LiveConnect in Netscape) is required. Netscape 4 and Internet Explorer 4 and 5 have been tested. A VRML viewer, such as CosmoPlayer 2.1 or higher, which works as a browser plugin and supports the proposed External Authoring Interface is required. GEIS itself requires a Perl 5.004 (or higher) interpreter and the Perl module XML::DOM to be properly installed to operate.

GEIS takes as input an XML/GEML file describing the documentary to be generated. XML provides a display, media, and platform independent form of information description and transfer. The format of the XML file is described by the GEML DTD. GEIS uses the GEML DTD to validate its input files. XML support in GEIS is implemented through the use of XML::DOM, a Perl module for building Document Object Model Level 1 compliant document structures (See <http://www.w3.org/TR/REC-DOM-Level-1/>), and NSGMLS, a SGML parser and validator (See <http://www.jclark.com/sp/nsgmls.htm>). The XML file specifies the PDB files to use, identifies the regions of the molecules, and lists the user-defined

actions that drive the documentary. The format of the XML file is described in Section 2.1 and an example is discussed in Section 3.

GEIS's output consist of several HTML files and a VRML file. GEIS's VRML generation of the 3D structure of molecules is partially based on code from PDB2VRML (See [http://reality.sgi.com/horstv\\_basel/pdb2vrml/](http://reality.sgi.com/horstv_basel/pdb2vrml/)). When viewed with a frames-capable browser, the documentary appears as a set of four panels (See Figure 2). The top-left panel contains a color-coded legend of the regions of the molecule(s). Each region name is hyper-linked to its description. The top-right panel is the largest of the four panels and houses the VRML display. The bottom-left panel presents a menu of items defined by the documentary author. Each item can be linked to a document to be displayed or a series of actions to be performed on the VRML world. This provides the viewer easy and immediate access to "highlights" or "points of interest" the documentary author wishes to mention. The bottom-right panel is used to display documents that are linked to by the VRML world, by the menu items listed in the bottom-left panel, or by the legend panel.

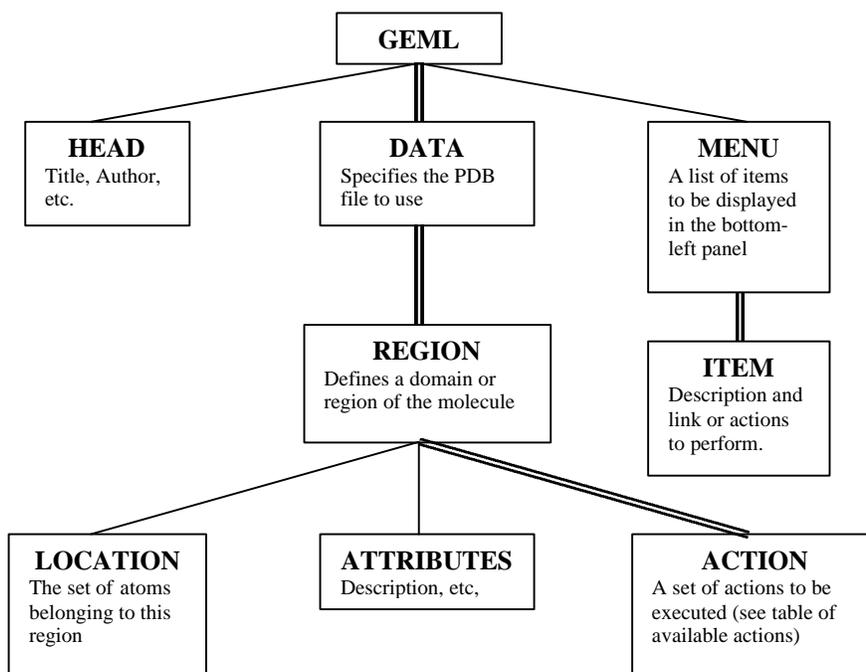


**Fig. 2:** A screen shot of the documentary frames. The content of the upper left window is derived from the HEAD and REGION definitions. The lower left window contains MENU items. The lower right window contains LINK descriptions, and the upper right contains the PDB structural data rendered according to the attribute choices.

### 2.1. GEML Input File

Our goal for the GEML format was for it to be an easily understood and logical hierarchy. The exact specification of the accepted format is provided by the DTD (See <http://www-smi.stanford.edu/projects/helix/geis/geml.html>). Figure 3 shows a tree representation of the tag hierarchy. The information used by GEIS is contained in a block enclosed by the tags <GEML> and </GEML>. Within the GEML block, three types of blocks can be specified: HEAD, DATA, and MENU. The HEAD block contains information about the documentary such as title and author. It can also contain user-defined key/value pairs using the META tag. The information provided in the HEAD block is used to generate titles and is inserted into the VRML file for documentation purposes. Much of this data appears in the upper left box of the documentary. The other two blocks, DATA and MENU, constitute the core specification of the documentary.

One DATA block is needed for each PDB file provided. The PDB file can be located on a local drive, on a web or FTP site, or can be loaded directly from the PDB. The following are all valid start tags:



**Fig. 3:** Schematic representation of the GEML XML format.  
(Double lines indicate multiple child nodes can be specified)

```
<DATA src="1a29.pdb">
<DATA src="http://smi.stanford.edu/~prasanth/1a29.pdb" other="on">
<DATA src="pdb://1a29">
```

The optional *other* attribute specifies whether atoms listed in the PDB file but not defined to be in any REGION should be rendered as a default “Other” region or not rendered at all. Each DATA block can in turn contain any number of REGION blocks. Each REGION block is used to define a separate domain or region of the molecule. The start tag appears as follows:

```
<REGION name="ReactiveSite" style="Wire" color="#FF0000">
```

Each region can be rendered in a different style. The currently available styles are Wire, Ball, Stick, BallAndStick, BallAndWire, CPK, and Surface. A color can also be specified for the region. A list of defined regions appears in the upper left box. The REGION block contains three additional types of blocks: LOCATION, ATTRIBUTES, and ACTION. The LOCATION block specifies the atoms, residues, or chains from the PDB file that constitute the region. The ATTRIBUTES block contains information that is used to generate HTML files for the documentary describing this region. Text enclosed by the DESCRIPTION start and end tags is copied directly to the output file, and generally appears in the relatively unconstrained lower right window. HTML tags can be embedded within the DESCRIPTION text as well. This permits the use of other plugins, applets, and tools not explicitly supported by GEIS. However, to prevent the HTML tags from being processed as XML tags, the text needs to be placed within the standard XML CDATA tag, as in the following example:

```
<ATTRIBUTES>
  <DESCRIPTION><![CDATA[
    <h1>Core - subdomain II</h1>
    <ul><li>crucial to catalytic activity (Redinbo, 1505)</li>
    <li>residues 233-319; 5 alpha helices and 2 beta strands</li>
    </ul>
  ]]></DESCRIPTION>
</ATTRIBUTES>
```

The ACTION blocks specify a set of actions. Each ACTION block can have a name and can be linked to an event in the VRML world (i.e. mouse over and mouse down). Named ACTION blocks can be called from other ACTION blocks using the PERFORM action described later in Section 2.2.

The MENU block consists of ITEM blocks that contain a menu item DESCRIPTION and a LINK (described below) or any number of PERFORMs. Each ITEM becomes a hyperlink in the lower left panel that executes the specified actions or link.

## 2.2. Actions

We examined a variety of scientific journal articles, describing molecular structures and cataloged the types of structural phenomena most frequently featured.

We associated a set of generic actions to the visualizations we found to be most often required. The goal of the actions was to permit the most common presentation requirements for structural articles to be met using GEIS's molecular documentaries. Table 1 lists the provided actions and the anticipated scenario of use for each action.

Actions act upon the region in which the ACTION block is defined. However, by specifying the optional *target* attribute, most actions can be made to act upon another region. Due to the complexity of most situations needing to be shown in a documentary, we added two additional actions: LINK and PERFORM, also listed in Table 1.

Our set of actions is designed to enable the creation of a wide range of documentaries covering papers describing many molecules and proteins. At the same time, we have kept the set of basic actions small enough to easily remember and quickly use. Through the use of the provided basic actions, complex higher level behaviors can be constructed in ACTION blocks.

The PERFORM action was designed to provide a method for creating reusable sequences of actions. The name of the ACTION block to execute, and the name of the region in which the ACTION is defined are required parameters for the PERFORM tag. In addition, the PERFORM tag takes an optional *delay* attribute which specifies the number of seconds to wait before executing the ACTION. The seconds are counted from the start of execution of the block in which the PERFORM lies. The following is a fragment of a block containing several PERFORMS which use the *delay* attribute to create an animation sequence:

**Table 1.** Provided actions

Action	Scenario of Use
BLINK	Draw attention to a region of the molecule
ACTIVATE	Highlight the region currently of interest
DEACTIVATE	Remove highlighting from a region when it is no longer the point of focus
TRANSPARENCY	Partially clear the view to see "inside" a molecule
ARROW	Point out regions, atoms, or locations of interest
MOVE	Animate regions or molecules
ROTATE	Rotate the entire world to see the molecule from another angle
SPIN	Spin a region of the molecule
ZOOM	Take a closer look at a region of the molecule in any direction or view the molecule from a different orientation
HIDE	Hide a region of the molecule for an unobstructed view of other regions
SHOW	Restore visibility of a previously hidden region
SHOWHIDE	Toggle a region between hidden and shown
RULER	Draw a line between two atoms and show the distance between them
<b>Additional Actions</b>	
LINK	Displays a HTML (or other browser-viewable format) file containing additional details, embedded sound or graphics, or another documentary
PERFORM	Separately define often used action sequences and then call them from other actions, eliminating data duplication

```

<PERFORM target="COOH_terminal_domain" action="moveOutC"/>
<PERFORM target="Core_Subdomain_I" action="moveInI" delay="8"/>
<PERFORM target="Core_Subdomain_II" action="moveInII" delay="14"/>
<PERFORM target="Core_Subdomain_III" action="moveInIII" delay="20"/>

```

GEIS is expandable to allow users to create their own actions by writing an associated Perl module. The module must be derived from `GEIS::Script`, the root object for all script tag handlers, and must implement two subroutines, *writePrototype* and *writeScript*. The source code for the provided actions can be examined and used as a template for custom actions. GEIS uses a separate Perl module to handle each action. Any action that has a tag defined in the DTD and a corresponding module in the scripts directory will be processed correctly. Thus the core GEIS code does not need to be modified in any way and third party modules can be distributed as add-ons. The source code for GEIS is available at <http://www-smi.stanford.edu/projects/helix/geis/source/>.

### 3. Case Study: Human Topoisomerase I

We selected two articles about human topoisomerase I to demonstrate the advantages and features of GEIS.<sup>7,8</sup> The two articles describe the same molecular structure, which is available as 1A36 in the PDB. The PDB file contains the enzyme complexed with a 22-base pair oligonucleotide. One article describes the structural features of the topoisomerase I enzyme and the second article presents a mechanistic description of the enzyme. We created a sample documentary to highlight points made in the articles. Our documentary can be viewed at <http://www-smi.stanford.edu/projects/helix/geis/documentaries/topo.html>. The GEML file used to create this documentary is available at [http://www-smi.stanford.edu/projects/helix/geis/documentaries/topo\\_geml.html](http://www-smi.stanford.edu/projects/helix/geis/documentaries/topo_geml.html).

The atoms of the topoisomerase I molecule were divided into regions, to define the critical molecular elements. GEIS can distinguish regions as sets of residue numbers for subunits and so we specified the residues corresponding to each of the units defined by the topoisomerase article authors. We defined eight regions: the linker domain, the COOH-terminal domain, three subunits of the core domain, and the DNA oligonucleotide. Two additional regions were specified to enhance the documentary: the amino acid residues of the active site and one other residue, Phe<sup>361</sup>. We used the “Wire” mode for most of the structure, but used the “BallAndStick” mode, a thicker mode, to make the active site more visible. Phe<sup>361</sup> was defined so that a ruler could be drawn across the molecule’s central pore. The text for the region descriptions that appears in the lower right panel is taken from the descriptions of the subunits found in the articles.

In this documentary, we used menu actions to provide general information and to visually clarify concepts. LINK actions were used to provide references and textual background for the molecule. We also utilized actions to toggle between

different forms of the 3D model, using the SHOW, HIDE, and SHOWHIDE actions. We also chose sections from the two articles that were difficult to visualize and understand, and created sets of actions for the documentary. For each menu item, we used combinations of all the actions to demonstrate concepts clearly by removing obstructive objects, and providing animations, labels, viewpoints, and descriptions. All the menu items in our documentary are described in Table 2. Text describing each menu item was placed into the body of LINK tags so that these pages are opened up at the same time as the other actions. By linking additional web documents, the documentary creator can add additional text or arbitrarily complex pages to include sounds, multimedia, or other web compatible technologies. For the "Examine active site" menu item, sounds were embedded into HTML documents, which also contain text descriptions. The LINK actions were grouped with the other actions so that the actions and the HTML documents appear simultaneously. The result is a voice narration through the entire action.

Many of the menu items are linked to several actions nested together. The combinations can be as simple as the "Label cleavage site" item, which consists of a label in the VRML world accompanied by a link to a hypertext document. The ARROW and LINK actions are grouped together in a single ACTION block.

```
<ACTION name="cleavageSite">
  <ARROW x="2" y="4" z="0" text="cleavage site"/>
  <LINK target="textbox" href="cleave.html"/>
</ACTION>
```

By combining many actions together hierarchically, we created menu items as complex as the "Examine active site" item, with changes in viewpoint, multiple labels, and multiple hypertext documents. In this menu item, actions which occur at the same time are grouped together into a single ACTION block. The first group is defined to contain actions to label an atom, zoom to the top of the active site, and open a hypertext document:

```
<ACTION name="active1">
  <ARROW to="2225" text="Arg 488"/>
  <ZOOM target="active_site" direction="top"/>
  <LINK target="textbox" href="active1.html"/>
</ACTION>
```

There are four such groups. These groups are then placed into the PERFORM tags in an ITEM block with appropriate delays. The delays are long enough for the sound files embedded in the hypertext documents to finish playing. The ITEM block is as follows:

```
<ITEM>
  <DESCRIPTION>Examine Active Site</DESCRIPTION>
  <PERFORM target="active_site" action="active1"/>
  <PERFORM target="active_site" action="active2" delay="7"/>
  <PERFORM target="active_site" action="active3" delay="14"/>
  <PERFORM target="active_site" action="active4" delay="30"/>
  <PERFORM target="Phe_361" action="rmArrow" delay="33"/>
</ITEM>
```

**Table 2.** Menu item descriptions

Menu Item	Description	Actions Used
1. References	HTML document with links to articles opens in text frame	LINK
2. Background	HTML document with background about topoisomerases opens in text frame	LINK
3. Show all	Shows all subunits	SHOW
4. Rotate world	Zooms around the entire world to view molecule from all sides	ROTATE
5. Assemble molecule	Separates the subunits of the enzyme from the DNA, spins the DNA molecule, and reassembles the subunits, with labels for each subunit	LABEL MOVE SPIN
6. N-terminal domain	HTML document with description of N-terminal domain opens in text frame	LINK
7. Toggle reconstituted molecule	Toggles the linker domain, HTML document with description opens in text frame	LINK SHOWHIDE
8. Toggle 63.4-kD particle	Toggles the DNA and linker domain, HTML document with description opens in text frame	LINK SHOWHIDE
9. Examine Active Site	Zooms in to the top of the active site; each residue is labeled and a separate HTML document is opened in text frame	LABEL LINK ZOOM
10. Examine Central pore	Zooms to the front of the COOH-terminal domain; toggles ruler for diameter of pore; toggles appearance of active site; toggles DNA; HTML document with description opens in text frame	BLINK HIDE LINK RULER ZOOM SHOW
11. Label Cleavage site	Toggles label of the cleavage site; HTML document with description opens in text frame	LABEL LINK
12. Subdomains I and II	Toggles label of the V between the subdomains; removes obstructing objects; zooms to three different viewpoints; at the final viewpoint; the objects are returned; HTML document with description opens in text frame	HIDE LABEL LINK ZOOM SHOW
13. Subdomains I and III	Zooms to front of DNA toggles label of the lip between the subdomains; removes the obstructing objects; HTML document with description opens in text frame	HIDE LABEL LINK ZOOM
14. DNA clamp	Repeatedly shows and hides Subdomains I and III; HTML document with description opens in text frame	HIDE LINK SHOW

For the “DNA clamp” menu item, we created a new blinking action by building upon the library of core actions that GEIS provides. The provided BLINK works for all rendering styles except Wire. The custom blinking behavior used by this menu item was created using repeated SHOW and HIDE actions instead. Two of the subunits blink to show how they contact with DNA by disappearing and reappearing again.

The “Assemble molecule” menu item is linked to a series of animations and spins. The animations were created from successive MOVE actions. We specified the coordinates and delay times for each step of the animation. In the following ACTION block for Subdomain I, the region moves back to its starting point from (40, -40, 0).

```
<ACTION name="moveInI">
  <ARROW text="Subdomain_I"/>
  <MOVE x="40" y="-40" z="0"/>
  <MOVE x="30" y="-30" z="0" delay="1"/>
```

```

    <MOVE x="20" y="-20" z="0" delay="2"/>
    <MOVE x="10" y="-10" z="0" delay="3"/>
    <MOVE x="0" y="0" z="0" delay="4"/>
  </ACTION>

```

The starting point for every region is at the origin; all the coordinates were calculated easily with respect to this common starting point.

Figure 4A in the article by Redinbo, et. al.<sup>7</sup> shows the topoisomerase molecule with a label of the cleavage site, but the exact location is unclear in a two-dimensional image. Our topoisomerase documentary contains a menu item to show a label in the 3D model, pinpointing the exact location of the cleavage site in the molecule. This label was determined using the textual description of the cleavage site's location. The user can also adjust the rotation and zoom to examine the cleavage site from different viewpoints, while the two-dimensional image is static.

By clicking on any region in the 3D model, the user can open an HTML document containing the description for that region. These VRML object links were created by assigning an event to the action. The following ACTION block, within the active site region, links to an HTML document:

```

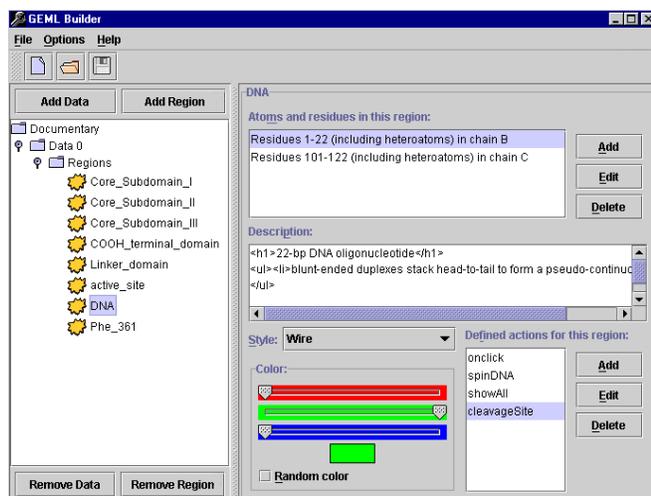
  <ACTION event="onclick">
    <LINK href="active_site.html" target="textbox"/>
  </ACTION>

```

When the user clicks on any VRML object that is part of the active site region, the link will be executed. Any type of action will work within an event driven ACTION block.

#### 4. Discussion

We have developed a prototype Java application, GEMLBuilder, for editing GEML files (available at the GEIS web site, see Figure 4). As XML becomes more accepted, interest is evolving in GUI applications to create and edit XML files (See <http://w3c.org/XML/#software>). Although GEML is a human readable format, having a GUI application to create and edit the XML file required for GEIS reduces the necessity of learning and understanding the specifics of the format. XML provides the ability explicitly to declare features of interest, independent of the mechanism for interpreting the XML. We have developed a core set of 3D interactions to describe molecules, which expand the limited descriptive capabilities of working within only two dimensions. These interactions are designed to cover the most frequently required methods for highlighting structural features in the macromolecular scientific literature. We used the implementation-independent XML format to describe this collection of functions and to create a system that allows 3D documentaries to be produced relatively easily without requiring detailed understanding of the documentary's underlying technologies or languages. The real long term benefit arises if our GEML DTD for describing features of molecules is an acceptable starting point for an open source molecular documentary language.



**Fig. 4:** Screen display of Java GEMLBuilder application. The left panel shows data blocks and regions. The upper right panel shows region definitions. The lower right shows region properties and a list of associated actions.

Other programs to process GEML can be developed to suit user and platform requirements, with no or little change in the underlying explicit declaration of the molecule's structure and function. The primary challenge in GEML is to identify a core set of actions that allow substantial control to the documentary author, but shield many of the details (and options) from them.

## References

1. F. C. Bernstein, T. F. Koetzle, G. J. Williams, E. E. Meyer, Jr., M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi and M. Tasumi, "The Protein Data Bank: a computer-based archival file for" *J Mol Biol* **112**, 3 (1977)
2. P. J. Kraulis, "MOLSCRIPT: a program to produce both detailed and schematic plots of protein structures" *J Appl Crystallogr* **24**, (1991)
3. R. A. Sayle and E. J. Milner-White, "RASMOL: biomolecular graphics for all" *Trends Biochem Sci* **20**, 9 (1995)
4. D. C. Richardson and J. S. Richardson, "The kinemage: a tool for scientific communication" *Protein Sci* **1**, 1 (1992)
5. G. Quinn, H. P. Wang, D. Martinez and P. E. Bourne, "Developing protein documentaries and other multimedia presentations for" *Pac Symp Biocomput* (1999)
6. A. D. Poon, L. M. Fagan and E. H. Shortliffe, "The PEN-Ivory project: exploring user-interface design for the" *J Am Med Inform Assoc* **3**, 2 (1996)
7. M. R. Redinbo, L. Stewart, P. Kuhn, J. J. Champoux and W. G. Hol, "Crystal structures of human topoisomerase I in covalent and noncovalent" *Science* **279**, 5356 (1998)
8. L. Stewart, M. R. Redinbo, X. Qiu, W. G. Hol and J. J. Champoux, "A model for the mechanism of human topoisomerase I" *Science* **279**, 5356 (1998)