

INTRODUCTION

Template Xplr2 is based on a simple frames webpage (i.e., frames5.htm). The "structure" webpage (i.e., pdbstructure5.htm) displays two *Chime*-rendered images. This template is the "workhorse" for structure exploration projects because it includes two **Command Entry Lines**, from which all *Chime/RasMol* commands can be issued, as well as two **Message Recall Boxes**, which continuously records every command issued. The Message Recall Boxes makes it much easier to "debug" commands by examining error messages or spurious information returned when an incorrect command is entered.

This template has been most thoroughly tested with the latest version of the *Chime* plug-in and the latest version of *MS Internet Explorer* running on a *Windows 98, 2000, or XP* operating system. Although the *Chime* plug-in seems to be most compatible with *Internet Explorer v6*, it also works with *Netscape* and *Mozilla* browsers running in a *Windows* environment. With *Mac OS* operating systems, *Netscape v4.7+* browsers may be the most reliable for *Chime* plug-in applications.

The following paragraphs outline a series of steps for revising these webpages using **Netscape Composer**, which is a free HTML editor, downloaded with **Netscape Communicator** (<http://channels.netscape.com/ns/browsers/archive47x.jsp>). You will need little knowledge of HTML ("Hypertext Markup Language") in order to customize these webpages, but note that webpages with embedded *Chime*-rendered 3D structures are more complicated than most basic webpages and therefore should be modified carefully and checked often in the browser after making revisions.

1. DOWNLOAD TEMPLATE XPLR2 ZIP FILE AND EXTRACT TO YOUR DESKTOP

Go to the "Demo & Downloads" page at the following URL:

<http://tutor.lscf.ucsb.edu/instdev/sears/biochemistry/presentations/demos-downloads.htm>.

Save the template zip file to your desktop by clicking on the link shown and selecting the "Save" option. Unzip/Extract the zip file to your desktop. This will create a new folder labeled "2structures-commandbox-template" and it will contain all the necessary files and webpages for this template.

2. TEST TEMPLATE XPLR2 IN THE BROWSER

Navigate to the folder created by the extracted zip file and double click on the **frames5.htm** webpage found in this folder. This will open the "frames" webpage into your default browser window (e.g. Internet Explorer configured with the *Chime* plug-in) as shown in Figure 1. Test this template by manipulating the structure while holding down the left mouse button and moving the mouse. If the image moves with the mouse, Template Xplr2 is ready to be customized.

3. FIND AND DOWNLOAD YOUR PDB FILES

Find the two PDB structure files of interest – for example, at the RCSB PDB (Protein Data bank) web site (<http://www.rcsb.org/pdb/>) with over 30,000 resolved structures to choose from. Download the files into the same folder created for the template.

Note the primary citations (journal article) for these structures and obtain these articles to help you prepare your presentation. Read through it and get a good idea of what is important and interesting about your structure. You may also want to locate other articles that give a broad background on your proteins' physiological function. You may want to ask questions like the

following about your structures: Are there any interesting diseases associated with the molecules? Do they bind a specific substrate of interest? And so on. Ultimately you'll want to compile a few paragraphs detailing the biological importance of your molecules.

4. POINT THE "STRUCTURE" PAGE TO YOUR PDB FILES

As a practical matter, the first step in customizing this template is to "point" the *Chime* plug-in to the new PDB structure file downloaded into the same folder as the template. Open the **frames5.htm** webpage with *Netscape Navigator (browser)*, click on the webpage, and select the *File/Edit Frame* pull-down option from the top menu bar (**Figure 1**), which will then open the **pdb-structure5.htm** webpage in *Netscape Composer*. Right click on the yellow tag for

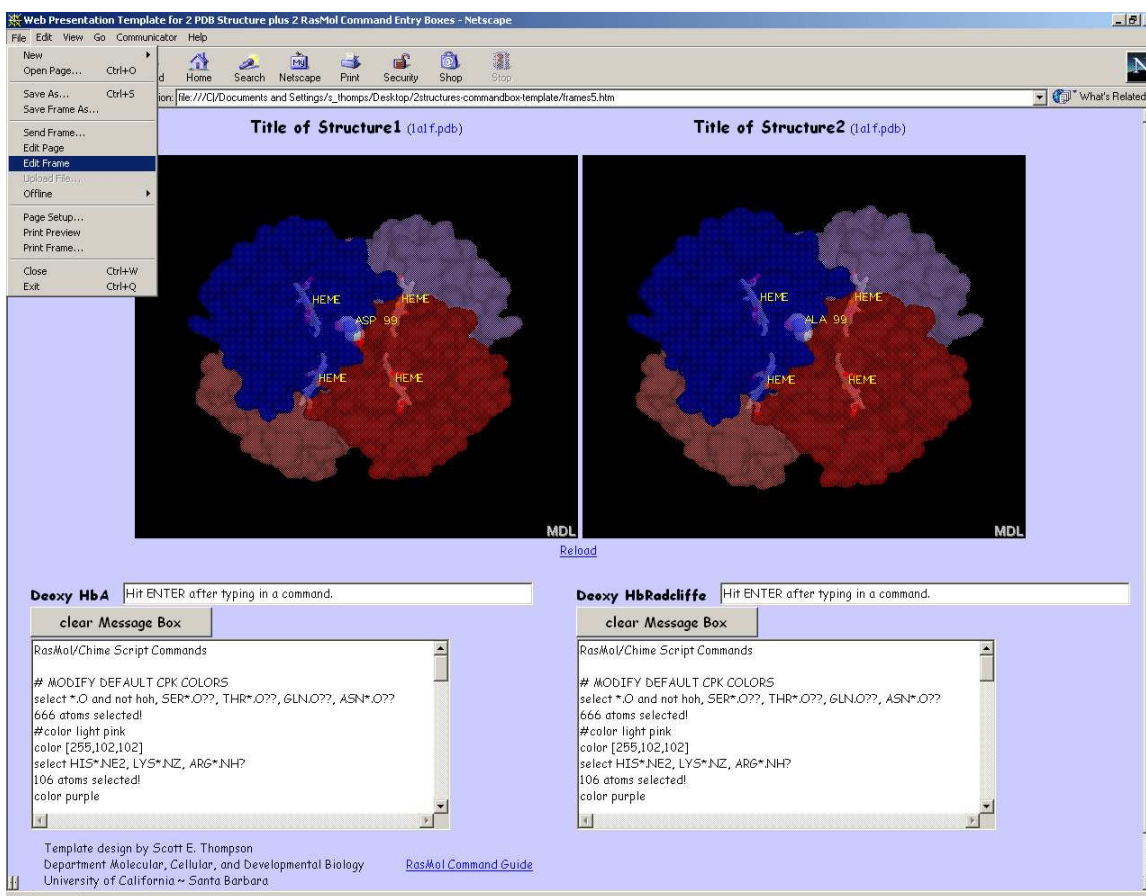


Figure 1. After clicking on the right frame, open the "structure" webpage in Netscape Composer by selecting the "File/Edit Frame" menu selection.

the *Chime* plug-in and select the *HTML Tag Properties* option, which will change the computer display to something like that shown in **Figure 2**, on the next page. Taking care not to delete all of the highlighted text in the open text box, click on the filename for the source tag (e.g. `src="filename1.pdb"`) and change the PDB file name to the new PDB file name (e.g. `src="filename2.pdb"`) like that shown in **Figure 3**, also on the next page.

Repeat this process for the second embedded structure. Click *OK* to close the text box and select the *File/Save* option from the top menu bar.

Customizing Template Xplr2 for Exploring 2 PDB Structures, Command Entry Lines, & Message Recall Boxes

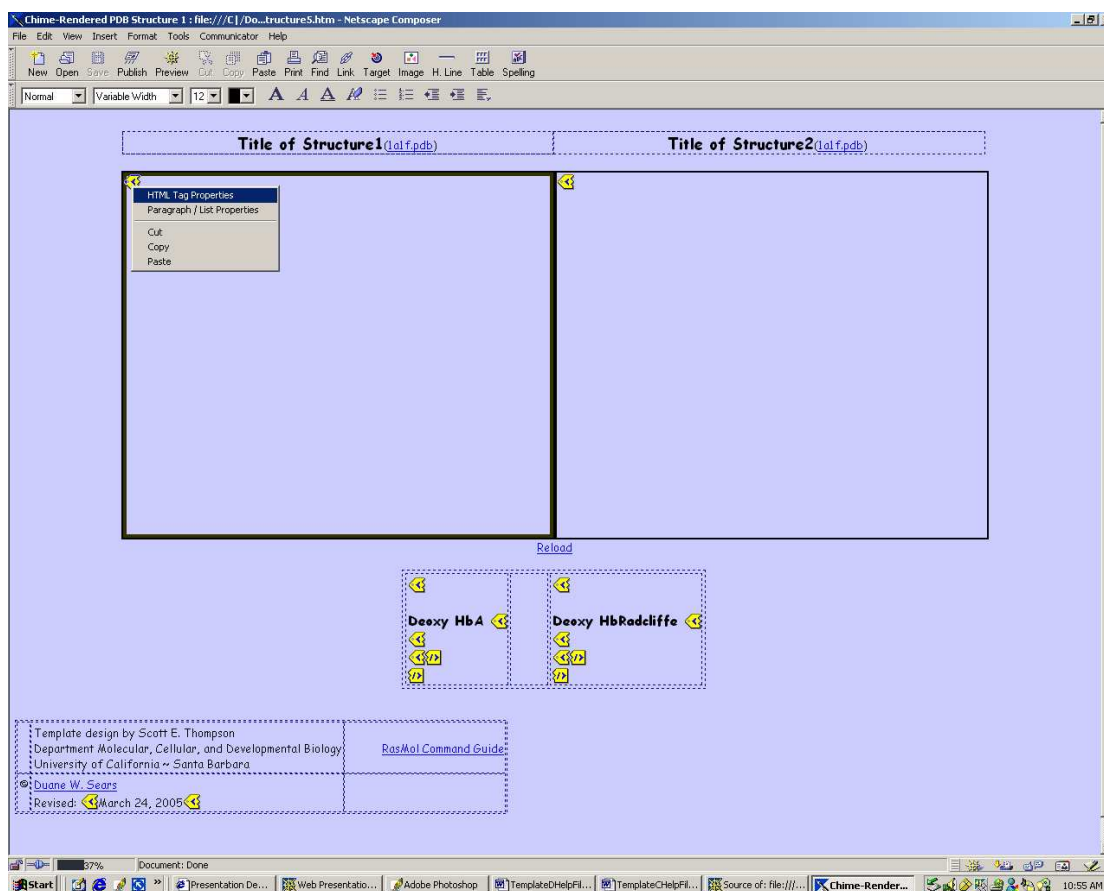


Figure 2. Access and edit the HTML Tag Properties for embedded structure.

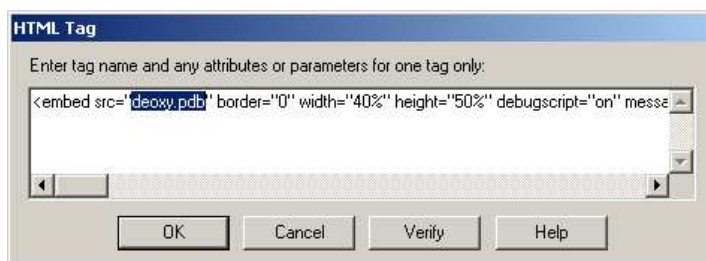


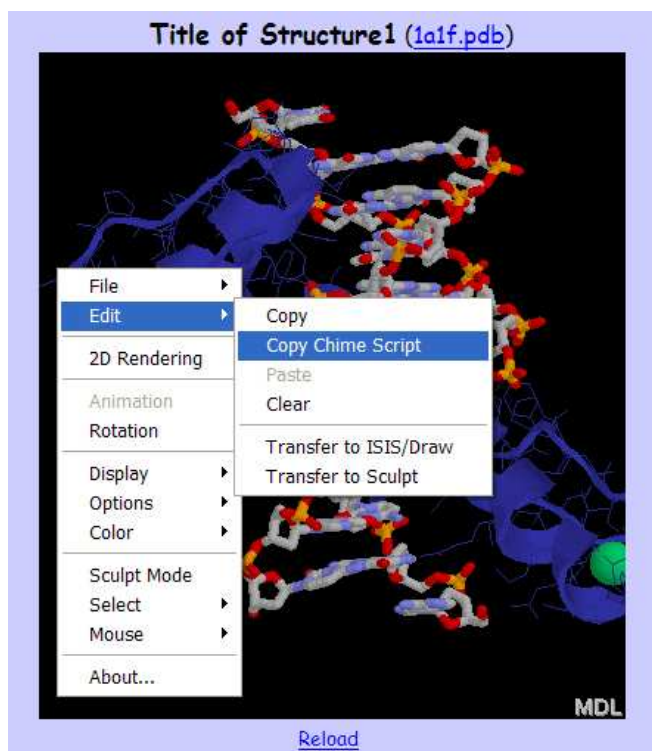
Figure 3. HTML Tag Window¹.

Toggle back to the browser window view and right click on the webpage in the right frame (but not on top of the structural image itself), and select *Reload* in order to refresh the image. If the structure corresponds to the new PDB structure, the template is ready for additional customization.

¹<embed src="deoxy.pdb" border="0" width="40%" height="50%" debugscript="on" messagecallback="ChimeMsgA" messages3d="true" name="structure5a" palette="foreground" script="# MODIFY DEFAULT CPK COLORS; select *.O and not hoh, SER*.O??, THR*.O??, GLN.O??, ASN*.O??; #color light pink; color [255,102,102]; select HIS*.NE2, LYS*.NZ, ARG*.NH?; color purple; select hoh; color magenta; select *; wireframe off; select asp99:d; spacefill; define ad (*a, *d); define dresidues (98,100-102) and *d and not (backbone and not *.ca); define aresidues (41-42,94,97) and *a and not (backbone and not *.ca); select hem.c??; # color dark grey; color [153,153,153]; select *a; surface asurface red; select *b; surface bsurface [204,153,255]; select *c; surface csurface [255,102,102]; select *d; surface dsurface blue; list selected transparent; select hem; wireframe 100; select hem.fe; label HEME; spacefill 200; select asp99:d.ca; label %n %r; color labels yellow; select all;">

5. CUSTOMIZE THE APPEARANCE OF YOUR STRUCTURES WHEN INITIALLY LOADED IN THE WEB BROWSER

The next recommended step for customizing Template Xplr2 is to find suitable x-, y-, and z-rotations for analysis of the new structures. Usually, the default orientation for most PDB structures – i.e., the structural orientation with no x-, y- or z-rotation parameters specified – will seldom correspond to the best orientation for examining the salient features of your structure. Even though it is possible to change all three parameters incrementally until the structure is properly re-oriented, this can be a tedious trial and error process and there is a much easier way to accomplish this. First, re-orient the structure with the mouse until it looks like you want it. Then, right click on the *Chime* image and select the *Edit/Copy Chime Script* pop-up menu option as shown in **Figure 4**.



```
#!RasMol -script
# File: \sfc.3
# Creator: Chime Version 2.0
```

```
set load check on
load pdb "1A1F_1~1.PDB"
background [0,0,0]
set ambient 60
set specular off
```

```
reset
slab off
rotate z -134
rotate y 64
rotate x -174
zoom 155
```

```
center all
etc., etc.
```

Figure 4. Edit/Copy *Chime* Script pop-up menu selection and corresponding script

This command copies (to the clipboard) the complete listing of script commands needed to render the image as it is being displayed. After pasting the listing into a text editor, like *Notepad*, find the grouping of x-, y-, and z-rotations, as well as the zoom parameter, as shown on the right side of **Figure 4**. Enter these numerical values in place of the default values found in the *Chime* embed tag (see **Footnote 1** on previous page). Alternatively, use the text editor to replace the carriage returns between commands with semi-colons (e.g., ***rotate z -134; rotate y 64; rotate x -174; zoom 155;***) and then paste this line of text over the default values in the *Chime* embed tag (see **Footnote 1**). Repeat this process for the second embedded structure. Save and check the modified webpage in the browser before continuing to make changes.

Now in similar fashion, customize the other commands in the *Chime* embed tag (i.e., `script="command 1; command 2;"`) to finish modifying the appearance of your structures. For guidance in command structure, see the [RasMol & Chime Command Guide](#) downloaded with Template Xplr2. Be sure to separate each command with a semicolon; precede non-command

comments with a pound (#) sign; and make sure all internal quotation marks are single rather than double quotes (e.g., ...; *label 'Ala16';...*) because double quotation marks are reserved for the beginning and end of the script command tags. Modifications can also be made to the other tags found within the *Chime* plug-in embed tag (e.g., *width="40%" height="50%"*) but familiarity with how the *Chime* plug-in is designed to work in the browser may be required.

6. DEFINING GROUPS FOR EASY USE IN YOUR PRESENTATION

Knowing how to group atoms will also help you create easier scripts for buttons. To do this, open the "structure" page, right-click on the embedded structure, and select the *HTML Tag Properties* option. Here, you may create defined groups for easy selection later on your "buttons" page. In order to define a group, you must place the following in the script: ***define [arbitraryname] [parameter];***. If you scroll to the right in the *HTML Tag Properties*, you will find three examples of defined groups (see **Footnote 1**).

The first of the three *define* commands above selected all atoms belonging to chains A and D and defined these atoms as "*ad*". So if one were to make a button that said, for instance: "***select ad; spacefill; color red;***" then those atoms would be spacefilled and colored red. The second defined key residues sidechains in chain D (98, 100-102) as "*dresidues*", while the third example selected these same residues in chain A and defined them as "*aresidues*". The names that you use to define the groups selected can be anything except for those that are pre-defined in *RasMol/Chime*.

7. GENERATING A RUNNING RECORD OF INFORMATIVE STATIC IMAGES

You should also keep a running record of static structural images while exploring the structures. Select the *Edit/Copy* pop-up menu selection that appears after right clicking the mouse directly on the image (as also shown in **Figure 4**). Any *Chime*-rendered image can then easily be pasted into another application, such *MS Word*, *MS PowerPoint*, *Adobe Photoshop*, etc. By saving these images, you will be able to keep track of important discoveries you make during each structure exploration session, and you will be able to incorporate informative static images directly into the webpages created for your presentation.

When attempting to copy and paste a *Chime*-rendered image with the *Edit/Copy Chime* pop-up menu option, one occasionally encounters the following problem with *Windows* operating systems. The *Windows Clipbrd.exe* program, which stores the contents of any copy command, sometimes fails to accept the copied image if the clipboard already contains text (as generated, for example, during a prior word processing exercise). In other words, existing text in the clipboard appears to "block" the image from reaching the clipboard. This problem is easily corrected by manually emptying the clipboard. For example, open the *Clipbrd.exe* program (likely found in the \system32\ within the *Windows* operating folder) and delete its contents. Now *Chime* images can be repetitively copied and pasted to different applications with the *Edit/Copy* selection without further need to empty the clipboard manually. Note that *Chime* images copied and pasted into *MS Word* documents and *MS PowerPoint* slides can be resized to smaller images of much high quality than found with *Adobe Photoshop*.

The reason for mixing static images in your presentation with dynamic *Chime* images is that it is difficult to grasp the detailed nature and nuances of specific structures when only dynamic images are shown. In other words, it is difficult to remember structural images that are continually in motion or are being continuously altered. Thus, static "snapshots" showing key structural features help one retain mental models of the complex macromolecular structures you are presenting.

8. INTERCOMPATIBILITY BETWEEN DIFFERENT TEMPLATES

All templates are designed so that two or more templates can be open at any one time in different browser windows. Thus, a student could open both Template Pres2 and Xplr2 in different browser windows, use the Command Entry Line of Template Xplr2 to develop customized scripts for the buttons, and then transfer these scripts to the HTML for the other template. It should be noted, however, that the *Chime* plug-in will not function properly if any single template is open in two or more browser windows at the same time, even if the templates have been redirected to different PDB files. Because all *Chime* actions target a dummy *name* designated in the "embed tag," the plug-in "acts confused" when two browser windows are open with *Chime* plug-ins having the same "target name." Specifically, any *Chime* initiated action (e.g., pressing a *Chime* button) will change only the first *Chime* image loaded into the browser, even if this action originates from the second browser window displaying the other *Chime* image. However, this problem can easily be corrected by modifying the HTML of one template as follows: (1) Change the *name* in the "name tag" of the *Chime* "embed tag" (see **Footnote 1**) of one template to a new name, e.g., change *name*="structure5a" to *name*="structure501a". (2) Change the *name* in the "target tag" (see **Footnote 2**) of every associated button to this new name, e.g., change *target*="structure5a" to *target*="structure501a". (3) Change the *name* in the "onsubmit tag" to the new name, e.g., change *onsubmit*="exec_script("structure5a")" to *onsubmit*="exec_script("structure501a")", being very careful not to remove or alter the HTML brackets around the name -- i.e., (" and "). Repeat this process for the other embedded structure and buttons.