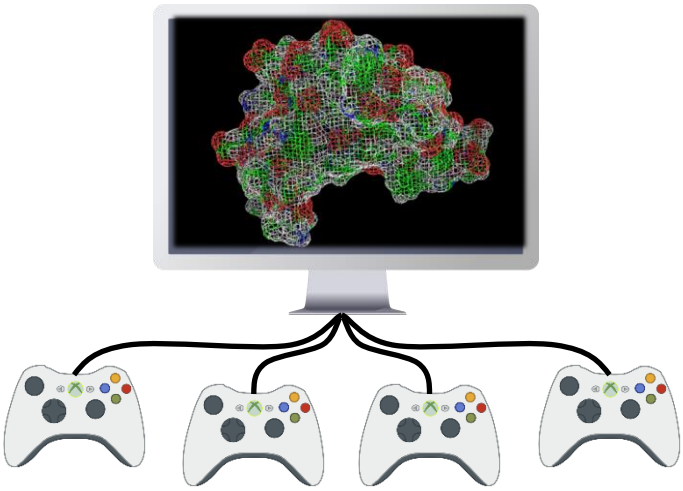
# PyMOL360 Guide



# Table of Contents

# Quick Reference

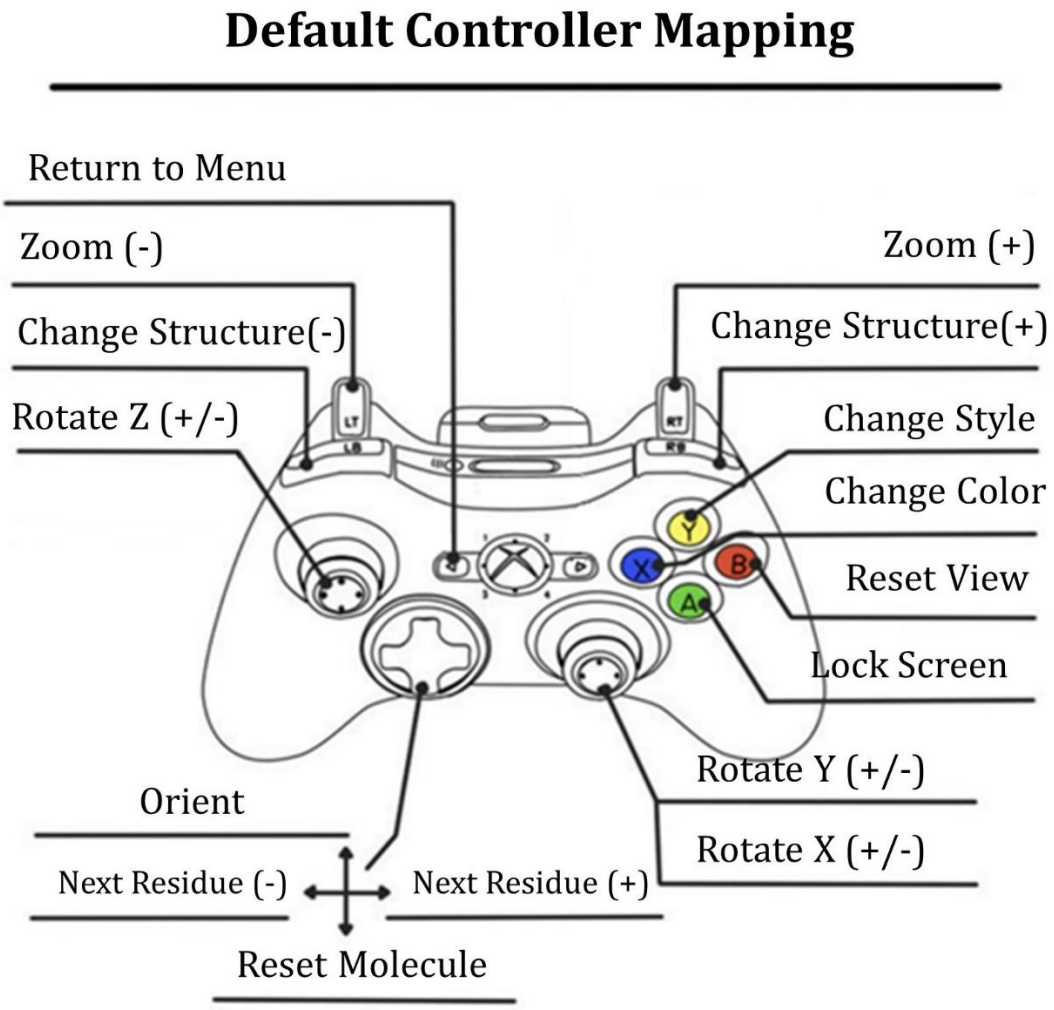



**Figure 1. Default controller mapping.**

# Walkthrough

## Contact Information

Welcome to PyMOL360! We are happy you are interested in using our plugin for PyMOL. If you have any trouble with installation or find bugs/error, please contact:

Ben Hackel
hackel@umn.edu
University of Minnesota

## Motivation

To help illustrate some of the capabilities of the PyMOL360 plugin, we will be exploring the active site of hen egg-white lysozyme enzyme. This molecule acts as a soluble antibiotic, hydrolyzing glycosidic linkages integral to bacterial cell walls. The crystallized structure can be accessed by the PDB accession code 3WVY. In addition to viewing the complete structure, we will highlight residues contributing to the electrostatic functions in this brief walkthrough.

## PyMOL Start-up

*Note*: Prior to starting this tutorial, make sure you have installed the Pygame module and downloaded the PyMOL360 script (PyMOL360.py). If not, see the README.txt for instructions.

1. Start your default PyMOL session by running the executable file. Resize the PyMOL viewer to a comfortable resolution

   *Note*: **Resizing the viewer** while PyMOL360 is running will **crash the application** due to an interaction between the Pygame module and the molecular viewer.
2. In either the Python terminal or the GUI command line, type "**run PyMOL360.py**". Be sure the file is in your current directory (this can be checked by typing "**pwd**" to *print the working directory* or "**ls**" to *list all files in the current directory*). By running this script, a new function, PyMOL360, is added to your PyMOL instance that activates gamepad control. This command must be repeated each time PyMOL is restarted due to session security (see http://pymolwiki.org/index.php/Extend).
3. To start PyMOL360, type "**PyMOL360**" into the PyMOL command prompt with a controller synced with the machine. A window should pop up to provide control over PyMOL360.

4. We are currently at the menu screen of PyMOL360. From here, we can modify which structures are active. Navigate using either the **keyboard arrow keys** (enter = select, esc = back) or the **gamepad directional keys** (A button = select, B button = back).
5. Enter the structures menu and turn off all structures. If **3WVY** is not listed, download it by selecting "Append Entry" and typing **"3WVY"** or **"3WVY.pdb"**.
6. Activate this structure by selecting the structure and turning it on by pressing the **A button** or the **enter key**.

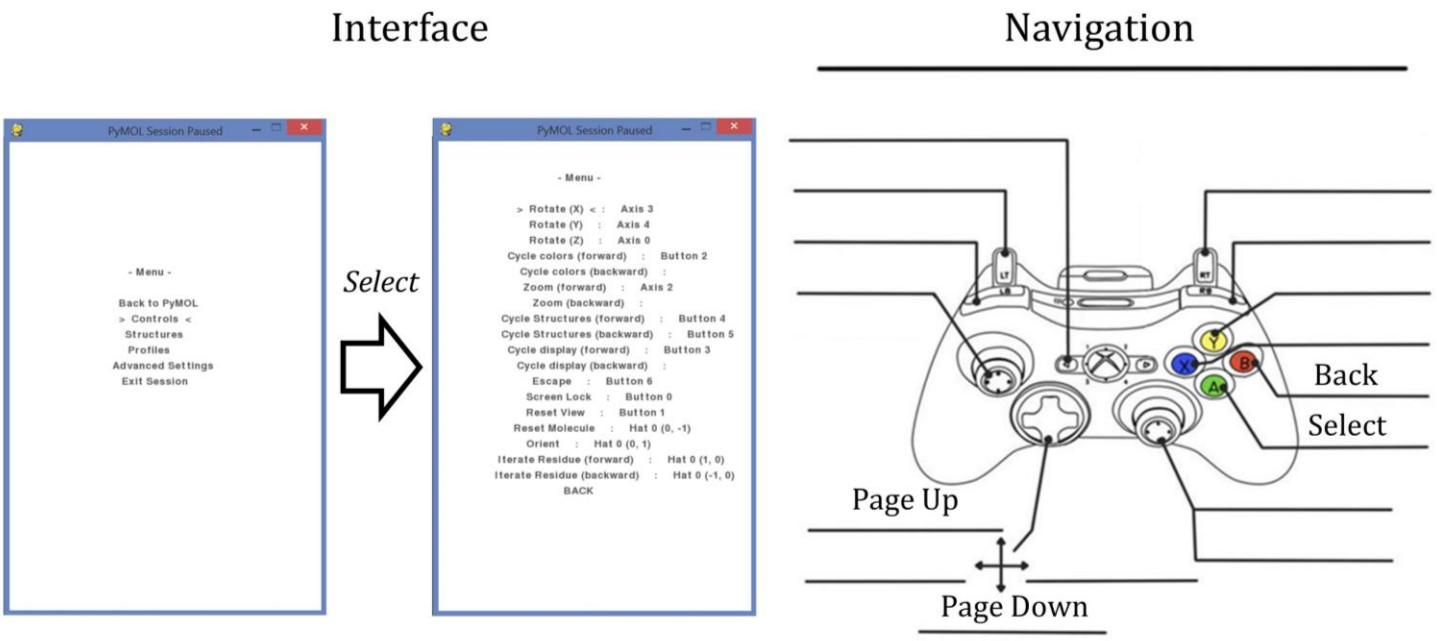



**Figure 2. Menu navigation.**

7. Return to the main menu by pressing the **B button**. Select "**Controls**" by navigating upward and selecting with the **A button**. It may be useful to review what each of these controls are mapped to by default, and change to your personal preferences. Above, a quick reference guide for Xbox 360 default controls is shown. If you would like to remap any controls, press the **A button** while highlighting a control and press the desired button. If you would like to clear the control (remove any existing control map), press **space** instead.
8. Return to the main menu by pressing the **B button**. Select "**Return to PyMOL**" to start the molecular visualizer for all selected molecules. If you want to quit at this point instead, you may select the **"Exit Session"** option which will close PyMOL360 safely. The remainder of this tutorial will explicitly highlight controls mapped through the

control defaults, although modifications can be made here to accommodate different controls and functions.

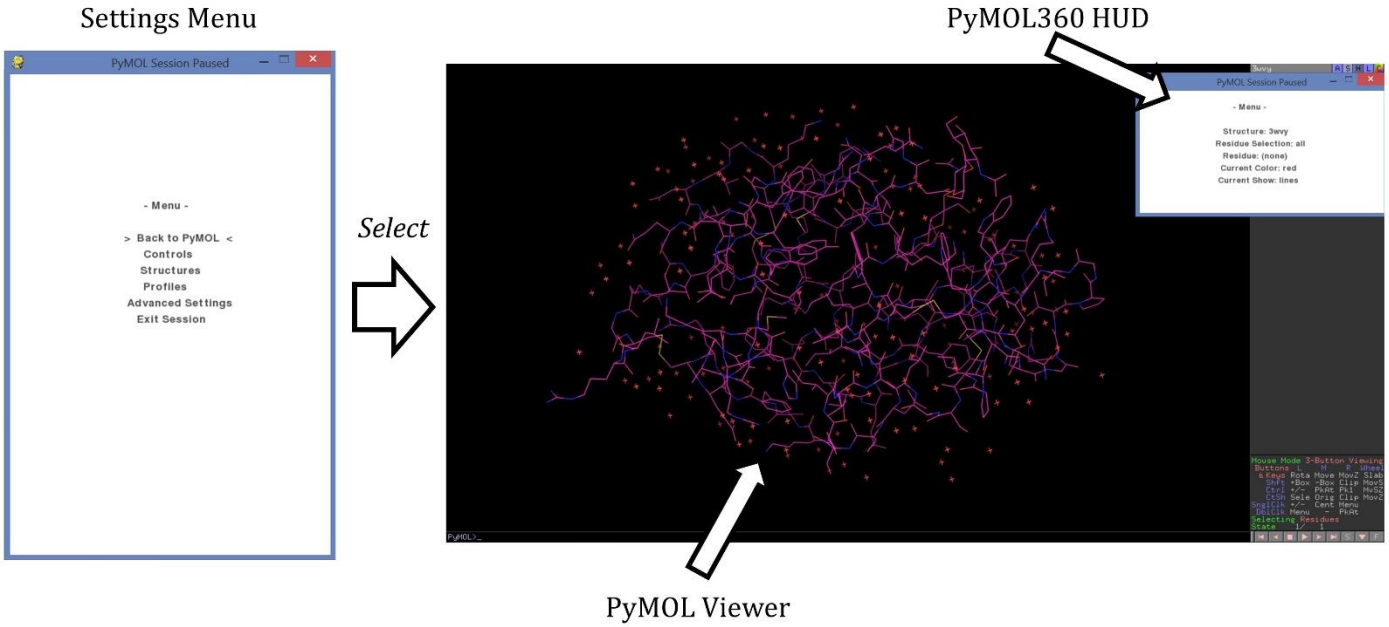



**Figure 3. Using PyMOL viewer with PyMOL360 heads-up display (HUD).**

# Structure Formatting

In this section, we will format the structure to help highlight the active region of the lysozyme molecule. At this point, the main PyMOL viewer should be open with a small secondary PyMOL360 window displaying information regarding the current structure and what is selected.

1. Let's start manipulating the view of our molecule:

   ***Rotation***: First, use the right joystick to control the x-direction and y-direction rotation of the active PyMOL structure. The left joystick can be used to rotate the molecule in the z-direction as well, if this manipulation is required.

   ***Zoom***: The left and right triggers control the proximity of the viewer to the molecule.

   ***Translation***: As an alternative to rotation, the structure can be shifted by locking the screen. This can be achieved by toggling the screen lock feature (**A button**). Once the screen is locked, the rotation controls (**right joystick**) now move the structure in the x-direction and y-direction.

***Reset View***: Returns your view to the original position upon loading. This can be completed by pressing the **B button**.

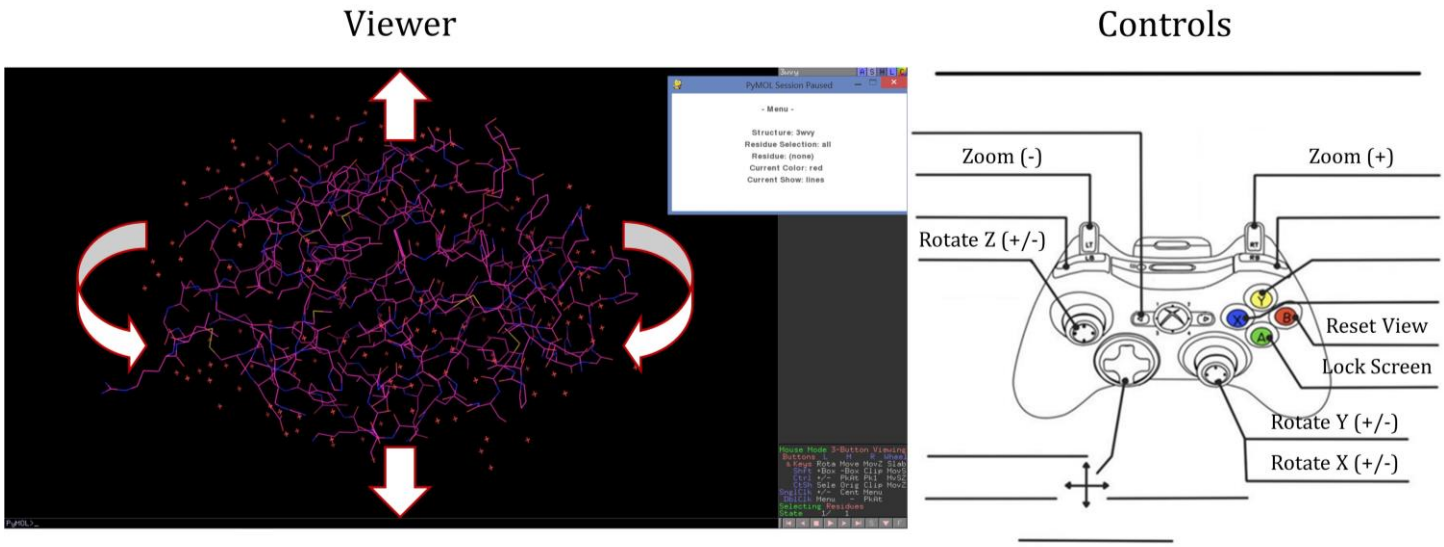



**Figure 4. Rotation, zoom, and viewing options.**

2. Once you feel comfortable adjusting the overall view of the molecule, we can start manipulating global properties of the structure:

   ***Change Color***: The color of the selection (by default, this will be the entire structure) will be changed. This can be cycled through by pressing the **X button**. The default colors are red, blue, green, teal, yellow, and white, but others can be added/removed in the advanced settings menu.

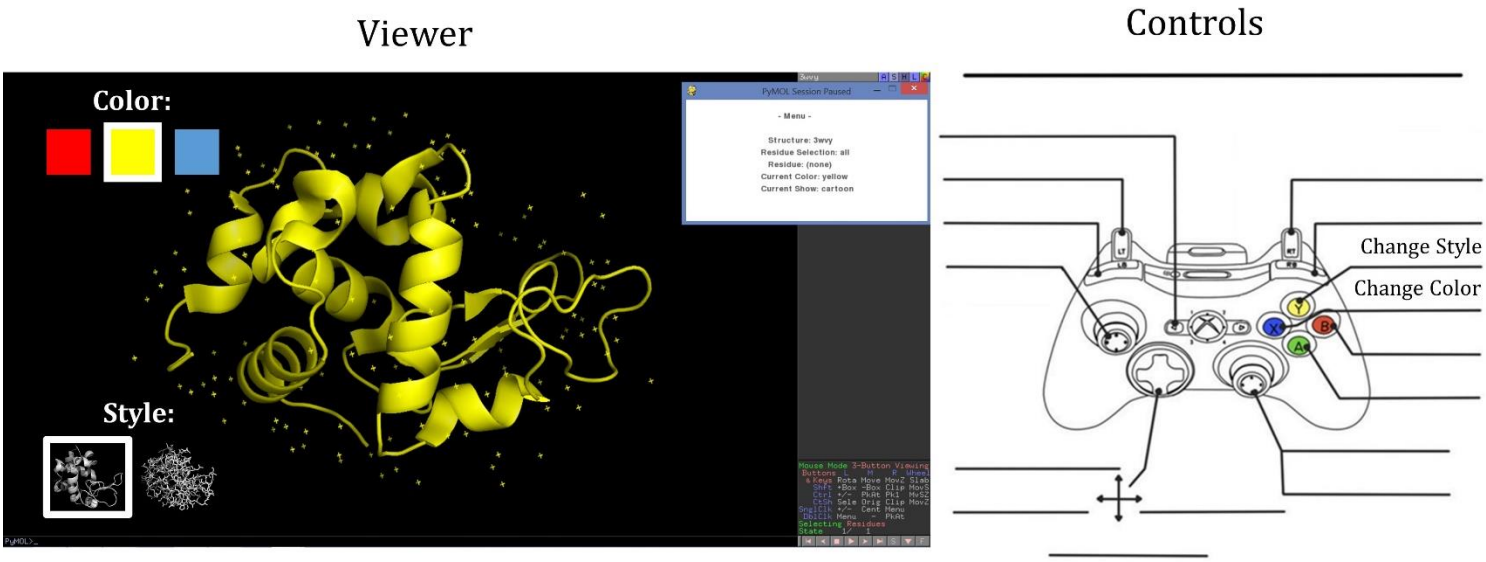



**Figure 5. Editing color and style.**

***Change Style***: In PyMOL, structures can be shown in a multitude of ways, including surface meshes, cartoons, and chemical stick and ball representations. The ability to switch between a subset of these forms can be achieved by pressing the **Y button**. The default styles are lines, sticks, cartoon, mesh, and dots, but other styles can be added/removed in the advanced settings menu.

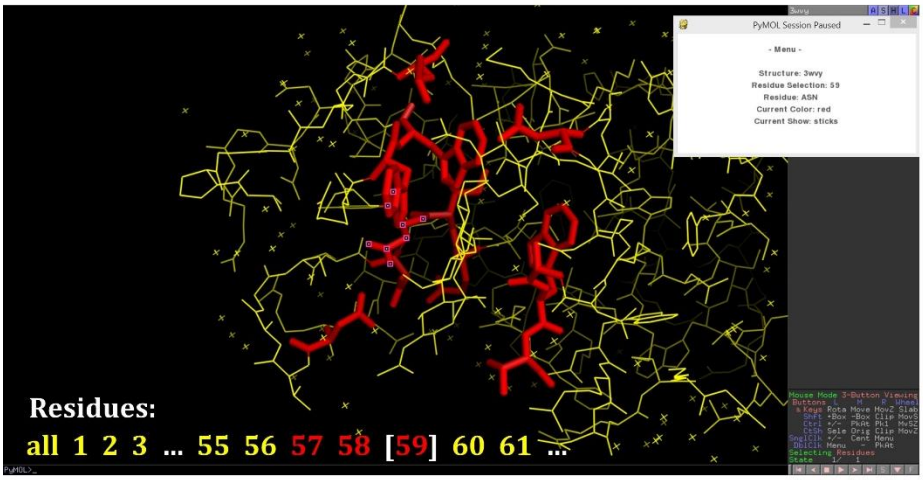
Use these functions to change the overall color of the structure to yellow (cycle via the **X button**) and the overall style to lines (cycle via the **Y button**).

3. So far, we have only shown ways to interact with the structure without specific control over individual residues of a structure. In the case of lysozyme (3WVY), we may want to highlight the particular chemical diversity of the active site of the enzyme. Next, we aim to format each of the residues that are active in ligand binding (residues ASN 46, ASP 52, GLN 57, ILE 58, ASN 59, TRP 62, TRP63, LEU 75, ASP 101, ALA 107, TRP 108, VAL 109).

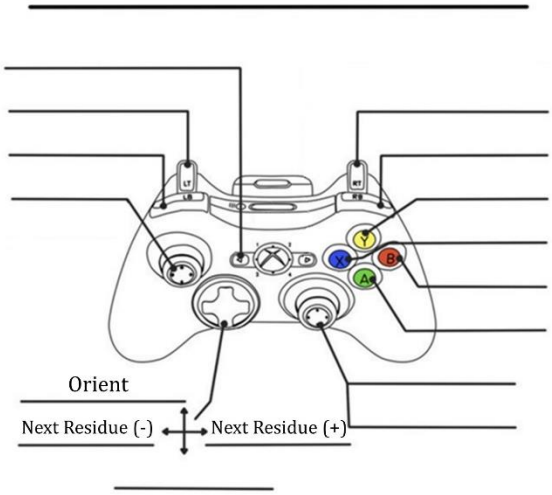
   ***Iterate Residue*:** The selection can be switched from the entire structure to individual residues in the structure by using the **left/right arrows** on the directional pad. The previously described functions of changing color and changing style will now apply to whichever residue is currently selected.

   ***Orient***: To focus the viewer on current selection, press the **up arrow** on the directional pad. When selection is all, the view of the structure is simply reset to its original position. However, when single residues are selected, the particular residue will be put immediately in frame. This is particularly helpful with larger structures were a residue is more easily iterated to that located in the molecular viewer.

Viewer




Controls



## Figure 6. Selecting and highlighting residues.

In practice, we can scroll through each of these residues until we reach an amino acid involved with ligand binding. To highlight to binding region, scroll to each residue (**left/right arrows**) and color each residue in the binding site to red and change the style to sticks (residues 46, 52, 57-59, 62-63, 75, 101, 107-109).

4. Consider a case in which there are a collection of structures you want to display in rapid succession. To manage this, let's return to the main menu:

   ***Return to Menu*:** To return to the main menu with all program settings. If selected, the current view of in the structure will be saved while navigating through the menu. This function can be accessed by pressing the **Back button**. Upon reselection of "Back to PyMOL" from the menu, the user will reestablish control of the molecular viewer.

   To add more structures to view at once in your PyMOL instance, return to the structures menu (Steps 3-6 of PyMOL360 setup). Now, activate any number of additional structures (press the **A button** when highlighting an inactivated structure). Press the **B button** to return to the main menu, then one more time to return to the PyMOL viewer. Now, we can switch to all activated structures rapidly:

   ***Change Structure***: This control will cycle through all activated structures. By default, we can activate this by pressing the **bumper buttons** on the top left and right of the controller. Each structure will store previous display settings when in an inactive form.

***Reset Molecule***: This final default control will reload the current selected structure, wiping all existing views and visual stylings. This feature will not affect disabled structures. To do this, press the **down arrow** on the directional pad.

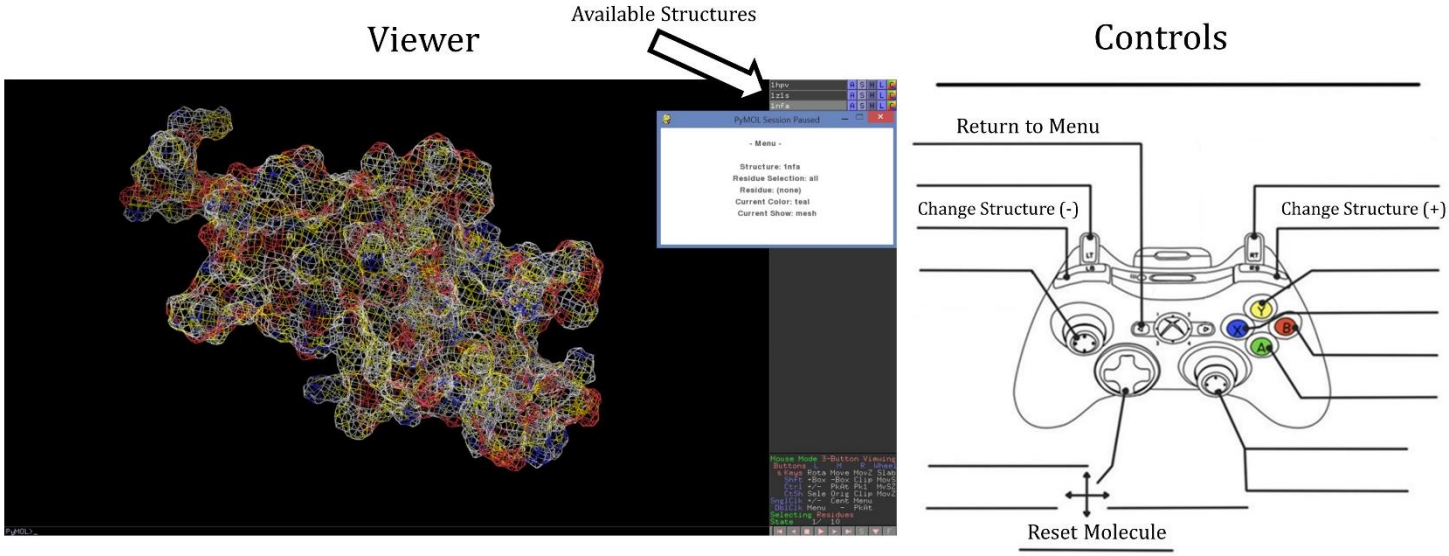



**Figure 7. Changing structures.**

5. These are all the basic controls for PyMOL360! Additional commands can be implemented via the instructions provided in the "PyMOL360.py" script. Although PyMOL360 can be closed by exiting the PyMOL360 window, this may lead to session instability, inadvertently closing the PyMOL interface as well. To exit and continue with the given PyMOL session, navigate to the "Exit Session" option in the main menu.

# Available Settings

As previously mentioned, all the settings of PyMOL360 can be accessed through the main menu. If we go here, we have four available options:

***Controls*:** Although this walkthrough is specifically geared towards use with an Xbox360 controller (or equivalent gamepad), we recognize there are many different types of joysticks available to users. Additionally, some functions may be less useful for some users, whereas others may need to add functions to accommodate their needs. In this menu, users may select any of the existing functions and map inputs from their controller.

***Structures*:** This menu was previously discussed in the "PyMOL360 Setup" section of the tutorial. From this submenu, users can control which structures are loaded locally and which structures will be activated to cycle through during usage. All shown structures are .pdb files found in the current direction for which PyMOL360 is placed.

***Profiles*:** This feature allows you to store specific configurations of settings under profile names. Moreover, in the next section, there will be a description of how to use this feature to save structures formatted prior to starting PyMOL360, allowing you to display a structure of any level of complexity while having an habitable platform to interact with the structure in real time.

***Advanced Controls*:** From here, you can control two separate features:

> *Colors*: Provides the ability to choose which colors you iterate through when using the "change color" function
> *Representation*: Provides the ability to change which styles you iterate through when using the "change style" function.

# Presetting Structures and Utilizing Profiles

For many demonstrations, it is heavily preferred all initial settings (controls, structures, colors, etc.) are preset so they can be accessed on the fly during a demonstration. This is accommodated through the "Profiles" section of the main menu. After adjusting any particular settings you are interested in, you can save the current setup as a profile. To do this:

1. Navigate from the menu to the "Profiles" options and select.
2. Select "Append entry" in this submenu and type (using the keyboard) a name for this setup.
3. Now, you can run the extension or the entire PyMOL instance. Upon subsequent starts of PyMOL360, you can return to the "Profiles" option and activate the profile to reacquire all the saved settings.

Additionally, we recognize that there is often no way to format a structure from the raw loaded or fetched form to a presentable object in a demonstration using only a few functions mapped to a controller. To compensate for this limitation, we provide the ability to save an existing formatted structure to a profile. To demonstrate, we will format the lysozyme structure, 3WVY, to a presentation ready form and save it to a profile:

1. The following arguments are only an example of how one could change the structure using controls not necessarily mapped in the control. This set of commands can be found on the PyMOL wiki gallery page. You can copy and paste this into the PyMOL command line:

```
fetch 1hpv
load set_color oxygen, [1.0,0.4,0.4]
set_color nitrogen, [0.5,0.5,1.0]
remove solvent
```

```
as spheres
util.cbaw
bg white
set light_count,8
set spec_count,1
set shininess, 10
set specular, 0.25
set ambient,0
set direct,0
set reflect,1.5
set ray_shadow_decay_factor, 0.1
set ray_shadow_decay_range, 2
unset depth_cue
```

2. Now that we have the structure formatted, we can save this particular view to a profile. Start PyMOL360 by typing:

```
PyMOL360
```

3. Now, you may modify any additional settings (i.e. controls, active structures, etc.) before saving this profile. In particular, it may be useful to turn off all structures other than "1hpv.pdb".
4. Navigate to the profiles menu and append a new profile. Let's call this particular file "1HPV_demo". Next time you start the PyMOL360, this profile will be present and will store the current view of the structure along with your setting preferences.

# Silent Start

When demonstrating, it may be of interest to bypass the interface altogether and immediately load the save file instead. In this case, from the PyMOL interpreter type:

```
PyMOL360 (save filename)
```

In the previous section, an example was given to generate a profile which we called "1HPV demo". If we wish to load this save without dealing with the interface, we can simply enter:

```
PyMOL360 1HPV_demo
```

The profile should be now loaded on PyMOL visualizer with controller compatibility. Note, the interface is disabled in this mode, so the **Return to Menu** function will exit the session instead.

# Acknowledgements

We would like to thank both Sadie Johnson and Ian Schrack for testing and feedback on the PyMOL360 package prior to release.