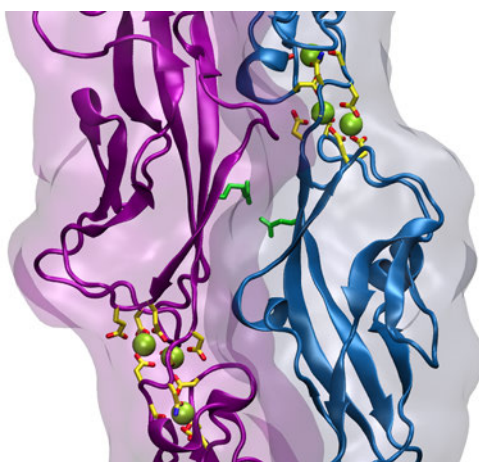




The Ohio State University
Department of Chemistry
and Biochemistry

Hands-on Introduction to Protein Visualization



VMD Developer: John Stone

Raúl Araya-Secchi

Michelle Gray

Marcos Sotomayor

October 2014

This tutorial is based on a VMD tutorial version created by Jordi Cohen, Elizabeth Villa, Marcos Sotomayor, and Klaus Schulten at the Theoretical and Computational Biophysics Group (UIUC), which is available at <http://www.ks.uiuc.edu/Training/Tutorials/>

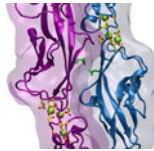
Contents

1	Basics of Protein Visualization in VMD	5
1.1	Loading Molecules	5
1.2	Displaying Molecules	6
1.3	Exploring Different Drawing Styles	7
1.4	Exploring Different Coloring Methods	10
1.5	Exploring Different Selections	10
1.6	Multiple Representations	12
1.7	Sequence Viewer Extension	13
1.8	Saving your Work	15
2	Multiple Molecules and Spatial Superposition	18
2.1	Loading Multiple Molecules	18
2.2	Using the VMD Main Window	19
2.3	Drawing representations for different molecules	20
2.4	Superposing molecules in space	22
3	Trajectories and Scripting	26
3.1	Loading Trajectories	26
3.2	VMD Main Window Animation Tools	28
3.3	Labels	30
3.4	Tcl Scripting Basics and the Tk Console	33
3.5	The <code>atomselect</code> Command	35
3.6	An Example Tcl Script: Molecular Separation in a Trajectory	38

Introduction

This tutorial introduces biochemists to protein visualization and analysis of molecular dynamics trajectories using VMD. *It is based on a version created at the Theoretical and Computational Biophysics Group at the University of Illinois at Urbana-Champaign. The tutorial is designed specifically for VMD 1.9.1 or later versions, and should take about 2 to 3 hours to complete in its entirety. Download and install VMD to start with tutorial instructions.*

The examples in the tutorial will focus on the study of two tip link cadherin proteins that mediate hearing. Throughout the text, some material will be presented in separate “boxes”. These boxes include complementary information to the tutorial, such as information about the biological role of “tip links”, and tips or shortcuts for using VMD.



Tip links. This tutorial will focus on the visualization of the *tip link cadherins* with VMD. Tip links are essential for hearing and balance, and are made of two proteins interacting tip to tip: cadherin-23 (Cdh23) and protocadherin-15 (Pcdh15). These are large proteins with long extracellular domains formed by extracellular cadherin (EC) repeats. The connection between Cdh23 and Pcdh15, the *tip link bond*, is mediated by the two most N-terminal EC repeats of each protein. The primary function of the tip link is to convey forces to the mechanosensitive channels that mediate our senses of hearing and balance.

Required programs

The following programs are required for this tutorial:

- **VMD:** Available at <http://www.ks.uiuc.edu/Research/vmd/> (for all platforms)
- **Plotting Program:** The very last part of the tutorial requires you to plot output from VMD using an external application. Some examples include:
 - Unix/Linux: xmgrace, <http://plasma-gate.weizmann.ac.il/Grace/>
 - Windows: Excel, <http://office.microsoft.com/en-us/FX010858001033.aspx> (Purchase required)
 - Mac/Multiple Platforms: Mathematica, <http://www.wolfram.com/> (Purchase required); gnuplot, <http://www.gnuplot.info/> (Free download)

Getting Started

You can find the files for this tutorial in the `vmd-tutorial-files` directory. Below you can see in Fig. 1 the files and directories of `vmd-tutorial-files`.

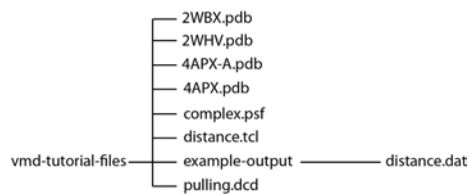


Figure 1: Directory structure of `vmd-tutorial-files`

To start VMD type `vmd` in a Unix terminal window, double-click on the VMD application icon in the `Applications` folder in Mac OS X, or click on the `Start` → `Programs` → `University of Illinois` → `VMD` menu item in Windows.

1 Basics of Protein Visualization in VMD

In this unit you will build a nice image of the Cdh23 EC1 protein fragment while getting used to basic VMD commands. In addition, you will learn how to look for interesting structural properties of proteins using VMD.

1.1 Loading Molecules

Our first step is to load our molecule of interest. A pdb file, `2WBX.pdb`, that contains the atom coordinates of the Cdh23 EC1 protein fragment is provided with the tutorial.

- 1 Choose the File → New Molecule... menu item (Fig. 2a) in the VMD Main window. Another window, the Molecule File Browser (b), will appear in your screen.
- 2 Use the Browse... (c) button to find the file `2WBX.pdb` in `vmd-tutorial-files`. Note that when you select the file, you will be back in the Molecule File Browser window. To actually load the file you have to press Load (d). Do not forget to do this!

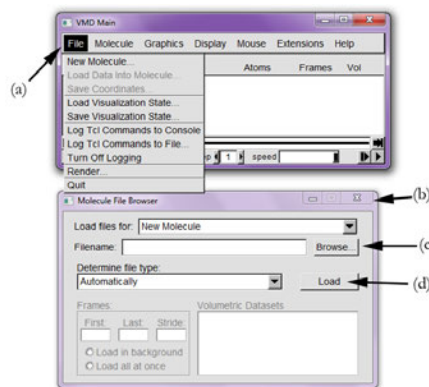
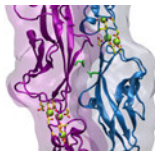


Figure 2: Loading a Molecule.

Now, the Cdh23 protein fragment is shown in your screen in the OpenGL Display window. You may close the Molecule File Browser window at any time.



Webpdb. VMD can download a pdb file from the Protein Data Bank if a network connection is available. Just type the four letter code of the protein in the File Name text entry of the Molecule File Browser window and press the Load button. For instance, you can type `2WBX` and VMD will download it automatically.



Coordinates file. The file `2WBX.pdb` corresponds to the X-ray structure of Cdh23 EC1 refined at 1.5 Å resolution reported by Marcos Sotomayor, Wilhelm A Weihofen, Rachelle Gaudet, and David P. Corey, *Neuron* (2010) **66**, 85. Note that the protein is surrounded by 105 water molecules, and that hydrogen atoms are not included.

1.2 Displaying Molecules

To see the 3D structure of the Cdh23 EC1 protein we will use the mouse in multiple modes.

- 1 In the OpenGL Display, press the first (left) mouse button down and move the mouse. Explore what happens. This is the rotation mode of the mouse and allows you to rotate molecules around an axis parallel to the screen (Fig. 3a).

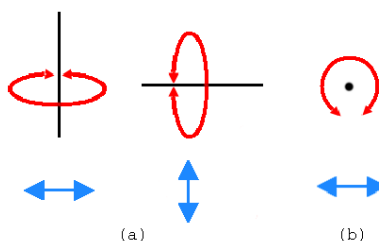


Figure 3: Rotation modes.

- 2 If you press the second (right) button and repeat the previous step, the rotation will be done around an axis perpendicular to your screen (b) (for Mac users, the second button is equivalent to hitting the command key while pressing the mouse button).

- 3 In the VMD Main window, look at the Mouse menu (Fig. 4). Here, you will be able to switch the mouse mode from Rotation to Translation or Scale modes.

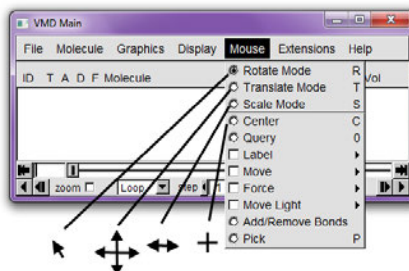


Figure 4: Mouse modes.

- 4 The Translation mode will allow you to move the molecule around the screen while holding the first (left) button down. While in translation mode, you can change the clipping plane by holding the mouse middle button down.

- 5 The Scale mode will allow you to zoom in or out by moving the mouse horizontally while holding the first (left) button down.

It should be noted that the previous actions performed with the mouse do not change the actual coordinates of the displayed atoms.



Mouse modes. Note that each mouse mode has its own characteristic cursor and its own shortcut key (r: Rotate, t: Translate, s: Scale) that could be used instead of the Mouse menu. Be sure to have the OpenGL Display window active when using the shortcuts. Additional information can be found in the VMD user's guide.

Another useful option is the **Mouse → Center** menu item. It allows you to specify the point around which rotations are done.

- 6 Select the **Center** menu item and pick one atom at one of the ends of the protein; The cursor should display a cross.
- 7 Now, press **r**, rotate your molecules with the mouse and see how they move around the point you have selected.
- 8 Select the **Display → Reset View** menu item to return to the default view.

The **Display** menu in the VMD Main window allows you to set multiple options controlling the way your molecules are displayed. Here you will switch depth-perception and rendering parameters.

- 9 Choose the **Display → Perspective** menu item.



Depth Perception. In perspective mode, things nearer the camera appear larger. The displayed image will not preserve scale relationships or parallelism of lines, with objects very close to the camera appearing distorted. The orthographic projection mode preserves scales and parallelism relationships, but greatly reduces depth perception.

Another way VMD can represent depth is through “depth-cueing”, which enhances three-dimensional perception but decreases the brightness of the scene.

- 10 Use the **Display → Depth Cueing** menu item to disable or enable this feature.

Finally, visualization of your molecule can be enhanced by using the **GLSL Render Mode**.

- 11 Select the **Display → Render Mode → GLSL** menu item.

If your computer supports **GLSL**, you should notice an improvement on the quality of the image displayed in the **OpenGL Display** window, specially in subsequent steps with more complex representations.

1.3 Exploring Different Drawing Styles

VMD can display your molecules using a wide variety of drawing styles. Here, we will explore those that can help you identify different structures in the protein fragment you loaded.

- 1 Choose the Graphics → Representations... menu item. A window called Graphical Representations will appear and you will see the current graphical representation used to display your molecule shaded (Fig. 5a).
- 2 In the Draw Style tab (b) we can change the style (d) and color (c) of the representation. In this section we will focus on the drawing style (the default is Lines).
- 3 Each Drawing Method has its own parameters. For instance, change the Thickness of the lines by using the controls on the right bottom part (e) of the Graphical Representations window.

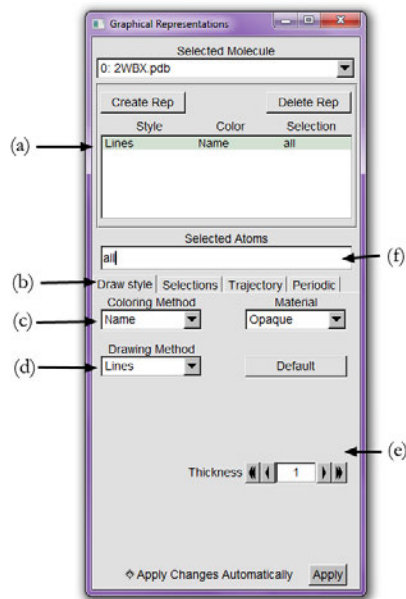



Figure 5: Graphical Representations window.

- 4 Now, choose VDW (van der Waals) from Drawing Method. Each atom is represented by a sphere. In this way you can see more easily the volumetric distribution of the molecule.
- 5 To see the arrangements of atoms in the interior of our protein, use the new controls on the right bottom part of the window (e) to change the Sphere Scale to 0.5 and the Sphere Resolution to 13. Be aware that the higher the resolution, the slower the display of your molecule will be. Changes in resolution will not be perceived if GLSL is active, but are important for image and movie rendering.
- 6 Note that in Coloring Method → Name, each atom has its own color, i.e: O is red, N is blue, C is cyan and S is yellow.
- 7 Press the Default button. This allows you to return to the default properties of the drawing method.



More representations. Other interesting representations are CPK and Licorice. In the first one, like in old chemistry ball & stick kits, each atom is represented by a sphere and each bond is represented by a cylinder (radius and resolution of both the sphere and the cylinder can be modified independently). The Licorice drawing method (widely used) also represents each atom as a sphere and each bond as a cylinder, but the sphere radius cannot be modified independently.

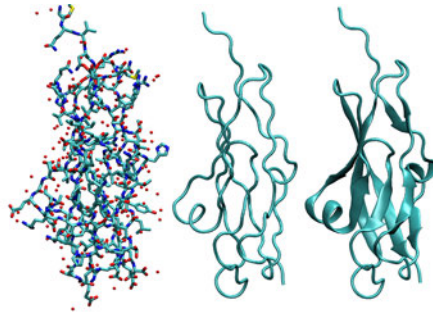


Figure 6: Licorice, Tube and NewCartoon representations of Cdh23 EC1.


The previous representations allow you to see the atomic details of Cdh23 EC1. However, more general structural properties can be seen by using more abstract drawing methods.

- 8 Choose the **Tube** style under **Drawing Method** and observe the backbone of your protein fragment. Set the **Radius** at 0.8.
- 9 By looking at your protein in the **Tube** mode, can you distinguish how many helices, β sheets and coils are present?

The last drawing method we will explore is **NewCartoon**. It gives a simplified representation of a protein based on its secondary structure. Helices are drawn as coiled ribbons, β sheets as solid arrows and all other structures as a tube. This is probably the most popular drawing method to view the overall architecture of a protein.

- 10 Choose **Drawing Method** \rightarrow **NewCartoon**.

You can now try to identify how many helices, β -sheets, and coils are present in this protein (Fig. 6).



Structure of a Cadherin EC fragment. Cadherin proteins have cadherin extracellular (EC) repeats with a typical fold and similar, but not identical, sequences. EC fragments or repeats are labeled EC1, EC2, and so on from N- to C- termini. Cdh23 EC1 has one short piece of 3_{10} -helix (residues 12 to 15), two turns of α -helix (residues 50 to 55), and seven β strands (residues 7 to 9 and 18 to 20 for strand A; 28 to 32 and 35 to 36 for strand B; 44 to 48 for strand C; 56 to 58 for strand D; 64 to 67 for strand E; 78 to 86 for strand F; and 91 to 100 for strand G). There are several reverse turns that link the strands together to form a Greek-key motif. VMD uses the program STRIDE to compute the secondary structure according to an heuristic algorithm.

1.4 Exploring Different Coloring Methods

- 1 Now, let's modify the colors of our representation. Choose Coloring Method \rightarrow ResType (Fig. 5c). This allows you to distinguish non-polar residues (white), basic residues (blue), acidic residues (red) and polar residues (green).
- 2 Select Coloring Method \rightarrow Secondary Structure and confirm that the New-Cartoon representation displays colors consistent with secondary structure (Fig. 7).

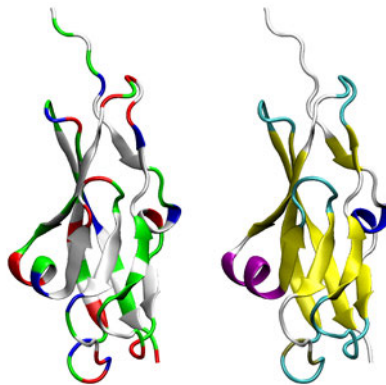


Figure 7: NewCartoon representation of Cdh23 EC1 colored by Residue Type (left) and Secondary Structure (right).

1.5 Exploring Different Selections

Let's look at different independent (and interesting) parts of our Cdh23 EC1 protein fragment.

- 1 In the Selected Atoms text entry (Fig. 5f) of the Graphical Representations window delete the word `all`, type `sheet` and press the **Apply** button or hit the Enter/Return key on your keyboard (do this every time you type something). VMD will show all β -strands forming sheets for `cdh23 EC1`. Note that β -strands A and B are split in two separate segments each.
- 2 In the Graphical Representations window choose the **Selections** tab (Fig. 8a). In section **Singlewords** (b) you will find a list of possible selections you can type. For instance, try to display α helices instead of β -strands by typing the appropriate word in the Selected Atoms text entry.

Combinations of boolean operators can also be used when writing a selection.

- 3 To see the molecule without helices and β strands, type the following in Selected Atoms: `(not helix)and(not betasheet)`
- 4 Now, change the current representation's Drawing Method to CPK style and the Coloring Method to ResName in the Draw Style tab.
- 5 In the section **Keyword** (c) of the **Selections** tab (a) you can see properties that can be used to select parts of a protein with their possible values. Look at possible values of the **Keyword** `resname` (d). Display all the Aspartates, Glutamates, and calcium ions present in the proteins by typing `(resname ASP)or(resname GLU)or(resname CA)`. Note that you can also type `resname ASP GLU CA`.

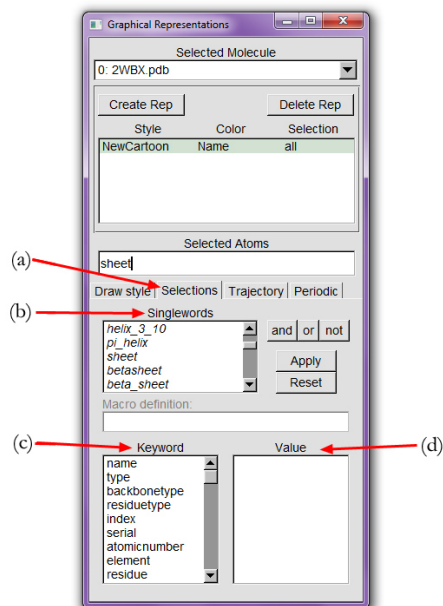


Figure 8: Graphical Representations window and the Selections tab.

Charged amino acids like Aspartate and Glutamate play an important role in calcium coordination and binding of Cdh23 to its Pcdh15 partner.

- 6 In the Selected Atoms text entry type `water`. Choose Coloring Method \rightarrow Name. You should see the 105 water molecules (in fact only the oxygens) present in our system.

7 To see which water molecules are closer to the protein you can use the command `within`. Type `water and within 2.6 of protein`. This selects all the water molecules that are within a distance of 2.6 angstroms of the protein.

8 Finally, try typing the following selections in Selected Atoms:

Selection	Action
<code>protein</code>	Shows the protein
<code>resid 1</code>	The first residue
<code>(resid 1 102) and (not water)</code>	The first and last residues
<code>(resid 50 to 55) and (protein)</code>	The α helix

All the previous options provide you with powerful tools to explore different parts of your protein.

1.6 Multiple Representations

The button `Create Rep` (Fig. 9a) in the Graphical Representations window allows you to create multiple representations. Therefore, you can have a mixture of different selections with different styles and colors, all displayed at the same time.

1 For the current representation, set the Drawing Method to `NewCartoon` and the Coloring Method to `Secondary Structure`.

2 In `Selected Atoms` type `protein`.

3 Press the `Create Rep` button (a). Now, using the menu items of the `Draw Style` tab and the `Selected Atoms` text entry, modify the new representation to get `Licorice` as the `Drawing Method`, `Name` as the `Coloring Method`, and `resname ASP GLU` typed in as the current selection.

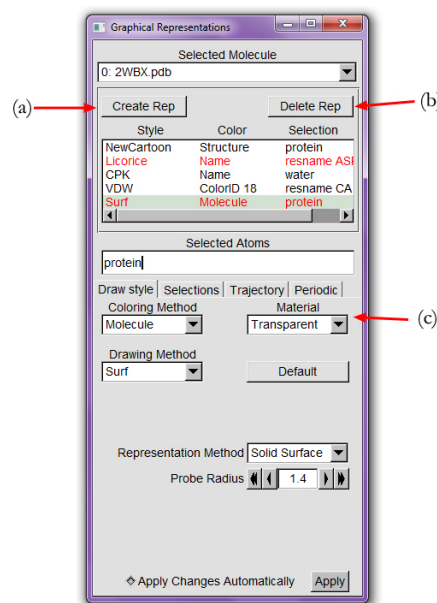


Figure 9: Multiple Representations.

- 4 Repeating the previous procedure, create the following two new representations:

Drawing Style	Coloring Method	Selection
CPK	Name	<code>water</code>
VDW	ColorID 18	<code>resname CA</code>

- 5 Create a final representation by pressing again the **Create Rep** button. Select **Drawing Method** → **Surf**, the **Coloring Method** → **Molecule** and type **protein** in the **Selected Atoms** entry. For this last representation choose in the **Material** section (c) the **Transparent** menu item.

Note that with the mouse you can select the different representations you have created and modify each one independently. Also, you can switch each one on/off by double-clicking on them or delete each one by using the **Delete Rep** button (b).

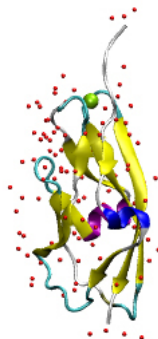


Figure 10: Cdh23 EC1.

- 6 Turn off the second and last representations. At the end of this section, the Graphical Representations window should look like Fig. 9, and your protein should look like Fig. 10.

1.7 Sequence Viewer Extension

When dealing with a protein for the first time, it is very useful to find and display different amino acids quickly. The sequence viewer extension allows you to pick and display one or more residues easily.

- 1 Choose the Extensions → Analysis → Sequence Viewer menu item. A window (Fig. 11a) with a list of the amino acids (e) and their properties (b&c) will appear in your screen.
- 2 With the mouse, click over different residues (e) in the list and see how they are highlighted. In addition, the highlighted residue will appear in your OpenGL Display window in yellow and bond drawing method, so you can visualize it easily. Use the right button of the mouse to unselect residues.

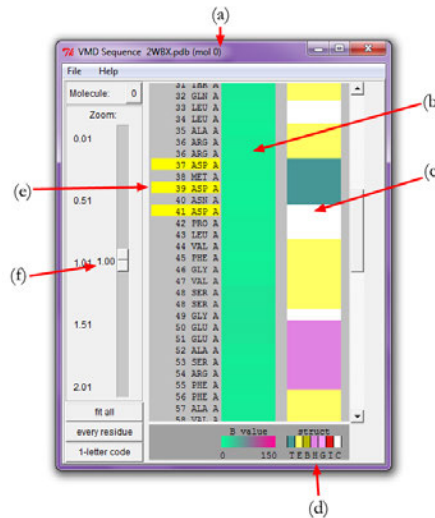
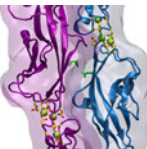


Figure 11: Sequence window.

- 3 Using the Zoom controls (f) you can display the entire list of residues in the window. This is especially useful for larger proteins.
- 4 Using the shift key while pressing the mouse button allows you to pick multiple residues at the same time. Look at residues 4, 5, 37, 39, 41, and 87 (e).
- 5 Look at the Graphical Representations window, you should find a new representation with the residues you have selected using the Sequence Viewer Extension. As you already did before, you can modify, hide or delete this representation.



Cadherins and Calcium. Cadherin proteins are involved in calcium-dependent molecular bonds. The calcium ion at the tip of Cdh23 EC1 clamps the N-terminus of the protein fragment to favor interactions with Pcdh15. Calcium is coordinated by side-chains and backbone atoms of residues 4, 5, 37, 39, 41, and 87 in EC1 (the ones that you selected using the Sequence Viewer Extension).

Information about residues is color-coded (d) in columns and obtained from STRIDE. The B-value column (b) shows the B-value field (temperature factor). The struct column shows secondary structure (d), where each letter means:

T	Turn
E	Extended conformation (β sheets)
B	Isolated bridge
H	Alpha helix
G	3-10 helix
I	Pi helix
C	Coil

Table 1: Secondary Structure codes used by STRIDE.

1.8 Saving your Work

The image that you have created using VMD can be saved, along with all representations you have created, as a VMD state. This VMD state contains all the information needed to start a new VMD session from it, without losing what you have done.

- 1 In the VMD Main window, choose the File \rightarrow Save Visualization State menu item. Write an appropriate name (e.g., `myfirststate.vmd`) and save it.

The File \rightarrow Load Visualization State menu item will allow you to load a previously saved VMD state, just like the file you saved. Although the VMD state allows you to work with the image and explore the properties of our protein using VMD, you usually need pictures that can be used in articles or other kind of documents. VMD can render the image you created and generate an image file that can be used in other applications, as it is shown in the following steps.

- 2 Using all that you have learned until now, find an appropriate view of the protein by scaling, rotating, and translating the molecule. Turn different representations on and off and improve the resolution and different properties of the selections you have made. If you want an image of high quality, put special attention to the resolution of each representation.
- 3 Be aware of the new representations you created with the Sequence Viewer Extension and hide or delete them if necessary.



Troubleshooting. If the directory path where you wish to save your rendered images contains spaces, it may be necessary, before you use the File Render Controls, to navigate to the directory you wish to place your file using the Tk Console (select the Extensions \rightarrow Tk Console menu item). Use quotes around the path, i.e., type `cd "C:\ Documents and Settings"`.

4 Before rendering the image, change the background color by choosing the Graphics → Colors menu item. There (Fig. 12), choose the Display category, the Background name, and the 8 white color. The background should be white now.

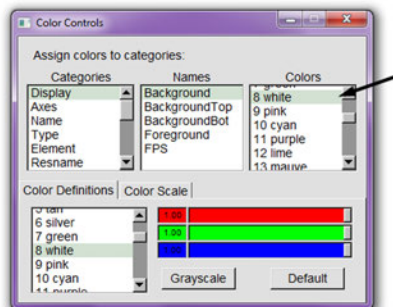
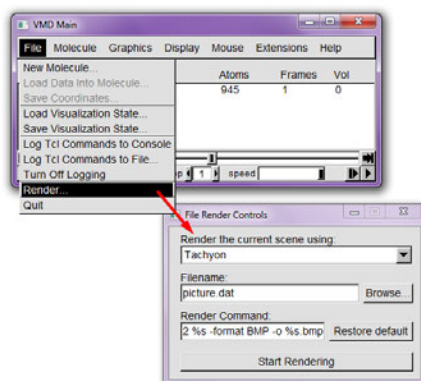


Figure 12: Color Controls window.

5 Choose the File → Render... menu item. A window called File Render Controls will appear in your screen (Fig. 13).

6 You can render the image using different packages. Pick TachyonInternal in the Render the current scene using menu if you are using Unix or Mac OS X, otherwise pick Tachyon.

7 Write the name of the file where the image will be saved in the Filename text entry, i.e., `picture.tga` or `picture.dat` (default is `plot.tga` or `plot.dat` depending on platform).



8 Press the Start Rendering button and the file with your image will be created. Note that this could take some time. You should end up with an image file named `picture.tga` (MacOS X or Unix) or `picture.dat.bmp` (on Windows).

Figure 13: File Render Controls window.

9 Close the application that opened the image file to continue using VMD (ignore this instruction in Windows).

10 In the VMD Main window, select the File → Save Visualization State menu item. Write an appropriate name (e.g., `myfirststate2.vmd`) and save it. This way you can reproduce your image later on if needed.

Now you are done with the first unit of the tutorial. We hope you have learned the basic commands of VMD. Also, you have generated at least two files. The first one is a VMD state that allows you to restart a VMD session, and use or modify all what you did in this unit. The second file is an image file of your protein that can be used in other image-viewing applications.



Advanced VMD features and tutorials. VMD is a powerful software with multiple tools that can help you generate sophisticated images and movies of your proteins. Here we have focused on the basics. The "Using VMD" and the "VMD Images and Movies" tutorials (<http://www.ks.uiuc.edu/Training/Tutorials/>) describe more advanced options and procedures that can help you take full advantage of VMD's capabilities.

2 Multiple Molecules and Spatial Superposition

In this second unit you will learn to deal with multiple molecules simultaneously within one VMD session. To do so, the structures of *Cdh23 EC1* and *Cdh23EC1+2* will be loaded and superposed. You will also load and spatially superpose the structure of *Pcdh15 EC1+2*.

Start with a new VMD session. If you have just completed unit 1, you should quit VMD and then launch it again.

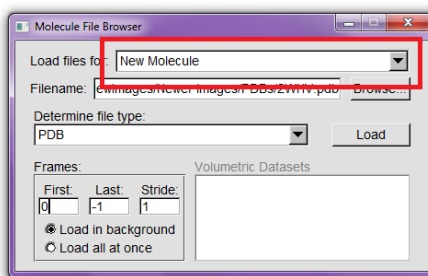
2.1 Loading Multiple Molecules

First, you will load some of the molecules needed for this session.

- 1 Open the Molecule File Browser from the File → New Molecule... menu item.

You need to load the X-ray crystal structure of *Cdh23 EC1*. You can do this in the same way you did in unit 1:

- 2 Use the Browse... button to find the file `2WBX.pdb` in `vmd-tutorial-files`. Note that when you select the file, you will be back in the Molecule File Browser window. In order to actually load the file you have to press Load. Do not forget to do this!

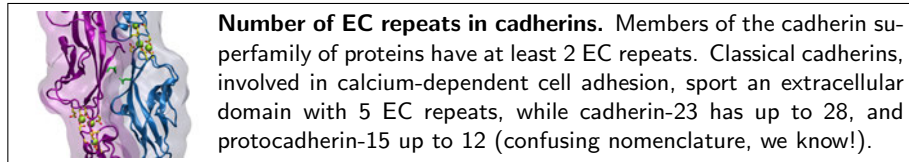


Now, the *Cdh23 EC1* fragment is shown in your screen in the OpenGL Display window. Figure 14: Molecule File Browser window with New Molecule option highlighted.

The X-ray crystal structure of *Cdh23 EC1+2* has been solved as well (pdb code 2WHV). Proceed to load it.

- 3 In the Molecule File Browser window, select New Molecule (Fig. 14) from the menu at the top of the window (next to Load files for:). Type 2WHV in the Filename: text entry and then press the Load button. If a connection to the internet is not available, browse for the file `2WHV.pdb` in `vmd-tutorial-files`, then click on the Load button. You have just loaded a separate molecule with the *Cdh23 EC1+2* structure.

You should now see two unaligned molecules. One is the structure of Cdh23 EC1, while the other one is the structure of Cdh23 EC1+2.



2.2 Using the VMD Main Window

You will now give names to your molecules so that you can identify them later.

- 1 Double-click on the name of the first molecule in the VMD Main window molecule listing. The Rename Molecule dialog box should pop up. Type in Cdh23 EC1. Do the same thing for the second molecule, and call it Cdh23 EC1+2.

At this point, your VMD Main window should look like Fig. 15. In front of the molecule name, there are four letters that you can use to manipulate your molecules.

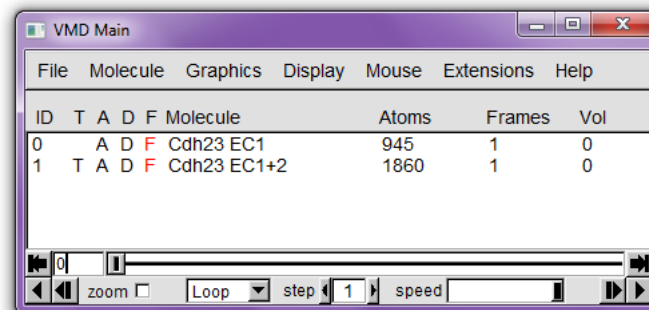


Figure 15: VMD Main window with Cdh23 EC1 and Cdh23 EC1+2 loaded into two separate molecules.

F stands for “Fixed,” meaning that the molecule won’t move when you move the scene around. When the F is black, the molecule is fixed; when it is red, the molecule is mobile.

- 2 Double-click on the F to the left of a molecule description in the Main form. Then, try translating the scene with the mouse while one molecule is fixed. Do it again with both molecules fixed.

- 3 When you are done, **unfix both molecules** and select the `Display → Reset View` menu item to correctly reposition the two molecules relative to each other.
- 4 Next, double-click on the D of one molecule. The D stands for “Displayed” and when the D is red it means that the molecule is hidden. You can control the visibility of all your molecules by double-clicking D.
- 5 When you are done, make sure that both the `Cdh23 EC1` and the `Cdh23 EC1+2` molecules are “Displayed” and that none of them are “Fixed”.
- 6 Finally, double-click under the T column to the left of the `Cdh23 EC1` molecule. The letter T should now appear in front of it. This makes it the unique “top” molecule. Making a molecule top makes it a target for multiple commands.

2.3 Drawing representations for different molecules

Before we continue and superpose these structures, take a look at your OpenGL Display window. You have two cadherin structures shown in the same default representation. To tell them apart, you can assign to them different representations and colors.

- 1 Open the Graphical Representations window by choosing the `Graphics → Representations...` menu item from the VMD Main window.
- 2 Make sure `0:Cdh23 EC1` is selected in the Selected Molecule pull-down menu on the top part of the Graphical Representations window. Select `Drawing Method → NewCartoon` and pick `Coloring Method → ColorID → 1` red for coloring.
- 3 Click on the Create Rep button to create a new representation. Delete the word `all` in the Selected Atoms text entry of the Graphical Representations window and type `resname CA`. Hit the Enter/Return key on your keyboard. Then select `Drawing Method → VDW` and pick `Coloring Method → ColorID → 18 yellow3` for coloring. You should be able to see one calcium ion displayed as a green sphere at the tip of the protein.
- 4 Click on the Create Rep button again to create a third representation that shows amino acids coordinating the calcium ion. Change the selection text to `same residue as within 3 of resname CA`, select `Drawing Method → Licorice`, and pick `Coloring Method → Name`.

One of the molecules has a particular set of representation and colors. Let’s now switch to the other one (Fig. 16).

- 5 In the Graphical Representations window, select 1:Cdh23 EC1+2 in the Selected Molecule pull-down menu on top. Select NewCartoon for the Drawing Method, and ColorID → 0 blue for Coloring Method.
- 6 Click on the Create Rep button to create a new representation. Delete the word **all** in the Selected Atoms text entry of the Graphical Representations window and type **resname CA**. Hit the Enter/Return key on your keyboard. Then select Drawing Method → VDW and pick Coloring Method → 18 yellow3 for coloring. You should be able to see **four** calcium ion displayed as yellowish spheres in Cdh23 EC1+2.
- 7 Click on the Create Rep button again to create a third representation that shows amino acids coordinating the calcium ions. Change the selection text to **same residue as within 3 of resname CA**, select Drawing Method → Licorice, and pick Coloring Method → Name.

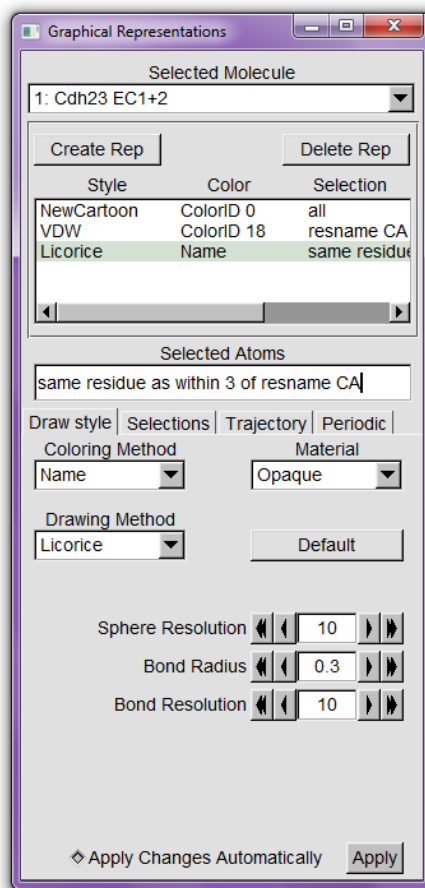


Figure 16: Graphical Representations window with 1:Cdh23 EC1+2 selected.

Note that calcium-coordinating amino acids are located at the tip of the protein, and also at the linker between repeats EC1 and EC2.

Now your OpenGL Display window should show Cdh23 EC1 colored in red and Cdh23 EC1+2 colored in blue, both with calcium ions shown as green spheres and with calcium-coordinating amino acids shown in licorice (Fig. 17).

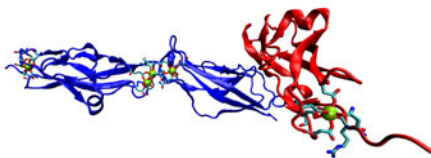
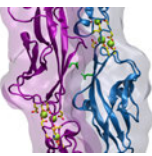


Figure 17: Cdh23 EC1 and EC1+2.

2.4 Superposing molecules in space

The two molecules shown in the OpenGL Display window have the EC1 fragment, yet they are not superposed in space so it is difficult to appreciate their similarities. We will use the RMSD Tool to superpose them in space.




Root Mean Squared Deviation RMSD. The Root Mean Squared Deviation (RMSD) is defined as:

$$RMSD = \sqrt{\frac{\sum_{i=1}^{N_{atoms}} (r_{iA} - r_{iB})^2}{N_{atoms}}} \quad (1)$$

where N_{atoms} is the number of atoms whose positions are being compared, and r_{iA} and r_{iB} are the positions of the corresponding atoms in molecules A and B , respectively. The RMSD can be used to superpose two similar structures by minimizing its value when applying rigid-body translations and rotations.

- 1 In the Main window, open the RMSD Tool window from the Extensions → Analysis → RMSD calculator menu item (Fig. 18).
- 2 In the text entry, delete `residue 5 to 85` and type `resid 5 to 85 and not altloc B C`. Leave the `Backbone only` box checked.

This creates a selection containing backbone atoms of amino acids 5 to 85 in both molecules, without including alternate conformations (`altloc`). This will ensure that the number of atoms being compared between the two molecules is the same, a requisite of the algorithm we use.



Alternate conformations. High resolution crystals structures often feature alternate conformations for amino acid side chains. Coordinates for all versions of an amino acid chain may be present in a single PDB file. VMD will show and use both, unless the `altloc` keyword is used to select for alternate conformations, usually labeled A to C.

- Click the **Align** button. You should be able to see in your OpenGL Display window how the two molecules get superposed, with the two EC1 fragments almost perfectly overlapping.

- Zoom in to appreciate the differences in the EC1 region.

Note that differences are most pronounced at the very N-terminal tips, which usually are fairly flexible and can adopt multiple conformations.

- Click the **RMSD** button to obtain a quantitative measure of the differences between the two EC1 structures.

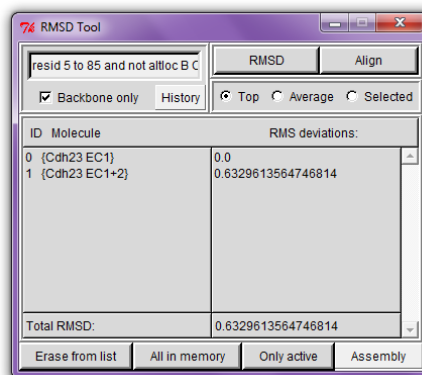


Figure 18: RMSD Tool window.

The comparison between EC1 fragments from these two crystal structures revealed minuscule differences, as expected. However, spatial superposition can help compare proteins that have similar, but not identical folds and structures. We will now load Pcdh15 EC1+2, and compare it to Cdh23 EC1+2.

- Open the **Molecule File Browser** from the **File** → **New Molecule...** menu item.
- Use the **Browse...** button to find the file **4APX-A.pdb** in **vmd-tutorial-files**. Remember to press **Load** in the **Molecule File Browser** window.

You should now be able to see three molecules in your OpenGL Display window.

- Double-click on the name of the third molecule in the **Main** window molecule listing. The **Rename Molecule** dialog box should pop up. Type in **Pcdh15 EC1+2**.

Pcdh15 EC1+2 is shown in the default **Lines** Drawing Method and spatially separated from the other two molecules. To change its representation and superpose it to Cdh23 follow these steps:

- Select the **Extensions** → **Visualization** → **Clone Representations** menu item. The **Clone Representations** window should pop-up. There, set the **From**

Molecule pull-down menu to 0 Cdh23 EC1 and the To Molecule: pull-down menu to 2 Pcdh15 EC1+2. Click on the Clone button. Pcdh15 EC1+2 should appear now in NewCartoon, with calcium ions in VDW and calcium-coordinating amino acids in Licorice.

- 10 In the Graphical Representations window, select 2:Cdh23 EC1+2 in the Selected Molecule pull-down menu on top, and click on the first representation to select it. Then change its color by selecting ColorID → 11 purple for Coloring Method.

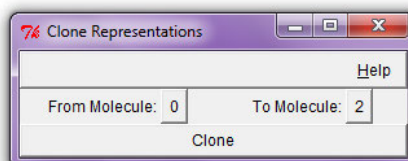


Figure 19: Clone Representations window.

You should be able to see Pcdh15 EC1+2 in purple, drawn with the same representations used for Cdh23 EC1 and Cdh23 EC1+2, but still spatially separated.

- 11 To superpose Pcdh15 EC1 to Cdh23 EC1, go to the RMSD Tool window and click on the All in memory button. This will update the list of molecules available for the RMSD Tool. All three molecules should be listed.
- 12 Check that the text entry still says resid 5 to 85 and not altloc B C and click on the Align button. Pcdh15 should now be superposed to the Cdh23 molecules (Fig. 20).

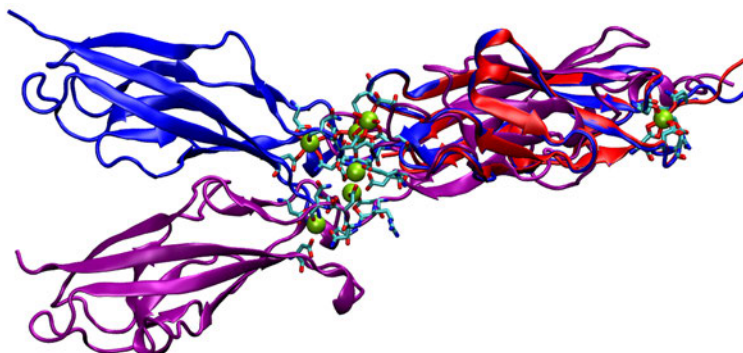
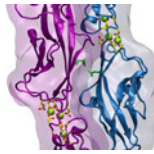


Figure 20: Cdh23 EC1 (red), Cdh23 EC1+2 (blue), and Pcdh15 EC1+2 (purple). All three molecules superposed using part of EC1 (residues 5 to 85).

The superposition is not perfect, but it is good enough to reveal a common fold for EC1 of both proteins. Note how the EC2 repeats are not aligned, as the angle between EC1 and EC2 repeats is different for Cdh23 and Pcdh15. Use all what you have learned about VMD to identify which parts of Cdh23 EC1 and Pcdh15 EC1 are similar to each other and which ones are not.



Cdh23 and Pcdh15 Tips. Cdh23 and Pcdh15 are proteins with long extracellular domains implicated in sound perception and deafness, as well as in blindness and cancer. The N-terminal tips of these proteins feature important structural features that are relevant for their function. For instance, both of them have long N-termini at their EC1 repeat, and large protrusions midway throughout β -strand A. These features are important for their binding as you will see in the next unit.

This ends the second unit of the tutorial. Now you should be able to handle multiple molecules in one session, use different representations and coloring methods for each, and superpose them to compare their structures effectively.

3 Trajectories and Scripting

In unit 3 you will learn how to load trajectories from molecular dynamics simulations, place labels on atoms and bonds, and use a simple script to calculate the separation between two molecules throughout a simulated trajectory.

3.1 Loading Trajectories

You will now learn to load the time evolving coordinates of a system, called *trajectories*. You will be able to see a movie of your system.

Trajectory files are normally binary files that contain several sets of coordinates for the system. Each set of coordinates corresponds to one *frame* in time. An example of a trajectory file is a DCD file. Trajectory files do not contain the connectivity information of the system contained in the protein structure files (PSF). So we first need to load the structure file and then add the trajectory data with the coordinates to this file.

- 1 Start a new VMD session.
- 2 Open the Molecule File Browser from the File → New Molecule ... menu item. Browse for the file `complex.psf` in `vmd-tutorial-files`, then click on the Load button. You have just created a new molecule with a structure (connectivity) but no coordinates (nothing will appear on the screen!).


Keep the Molecule File Browser window open. Notice that the menu at the top of the window now says 0: `complex.psf`. This ensures that the next file that you load will be added to that molecule (molecule ID 0). You do **not** need to change this menu to `New Molecule`, since you are merging data with the PSF file.

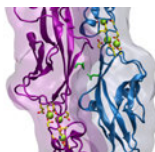


Coordinate files and structure files. To save space, simulation output files usually contain only the atom coordinates and do not store unchanging information such as the atom types, atom charge, segment names and bonds. The latter information is stored in a separate “structure” file (*e.g.*, a PSF file). To visualize the results of a simulation, you need to merge both a structure and a coordinate file into the same molecule.

- 3 In the Molecule File Browser window, click on the Browse button. Make sure that `complex.psf` is selected on the menu. Browse for `pulling.dcd` in `vmd-tutorial-files`, click OK, and click on the Load button again. You will be able to see the frames as they are loaded into the molecule.

After the trajectory finishes loading, you will be looking at its last frame. We will now create representations that will help you look at the system more easily.

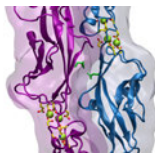
- 4 In the VMD Main window, click  to go back to frame 0. Then choose Graphics → Representations... menu item to open the Graphical Representations window. In the Selected Atoms text entry window, type `segname P1`. Then select Drawing Method → NewCartoon and Coloring Method → ColorID → 23.
- 5 In the Graphics → Representations menu item, click on the Create Rep button.
- 6 In the Selected Atoms text entry window, erase the text and type `segname P2`. Then select Drawing Method → NewCartoon and Coloring Method → ColorID → 11.
- 7 In the Graphics → Representations menu item, click on the Create Rep button to create a third representation.
- 8 In the Selected Atoms text entry window, erase the text that appears there and replace it with `resname CAL`.
- 9 In the Draw Style tab, choose a VDW representation for this selection and color it yellow (ColorID 17). You should now see all 7 calcium ions present in the system.



Cdh23 and Pcdh15 “Handshake” Complex. The tips of Cdh23 and Pcdh15 have been shown to interact through an extended “handshake” bond essential for tip link function. The structure, solved at 1.65 Å resolution, was also used for molecular dynamics simulations testing the mechanical strength of the bond (Marcos Sotomayor, Wilhelm A Weihofen, Rachele Gaudet, and David P. Corey, *Nature* **492** 128, 2012). Tip links are constantly under sound-induced tension, and unbinding of the complex is likely related to temporary deafness due to loud sound. The trajectory you just loaded is a simulation that mimics physiological stretching while hearing a very loud sound.

- 10 Create a representation with the selection `betasheet and backbone`, choose Drawing Method → Hbonds, and color it red selecting Coloring Method → Color ID. In the options, set Distance Cutoff to 3.2, Angle Cutoff to 30 and Line Thickness to 5.

In the OpenGL Display window you should be able to distinguish and appreciate the most important features of the handshake complex, including Cdh23 EC1+2 (blue), Pcdh15 EC1+2 (purple), calcium ions (green), and β -sheet hydrogen bonds (red). Your system should now look similar to the one in Fig. 21.



Cadherin Mechanics. The elastic properties of cadherins depend on calcium ions and hydrogen bonding between residues in the protein's β strands. Calcium rigidifies the chain of EC repeats, also preventing mechanical unfolding. Hydrogen bonds between β sheet backbone atoms give stability and their formation and rupture under mechanical stress confers EC repeats with their elastic properties.

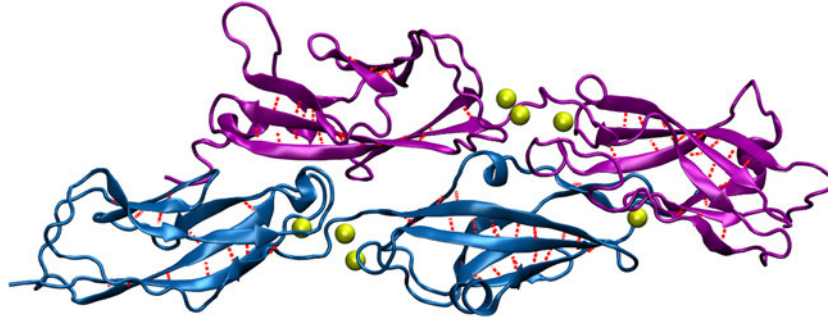


Figure 21: Cdh23 and Pcdh15 complex.

3.2 VMD Main Window Animation Tools

Now that you have nice representations, you will be able to observe what happens during your trajectory. The Animation Tools help you do that. The VMD Main window includes all the Animation Tools you need for navigating through your trajectories. These tools are located at the bottom of the window (Fig. 22).

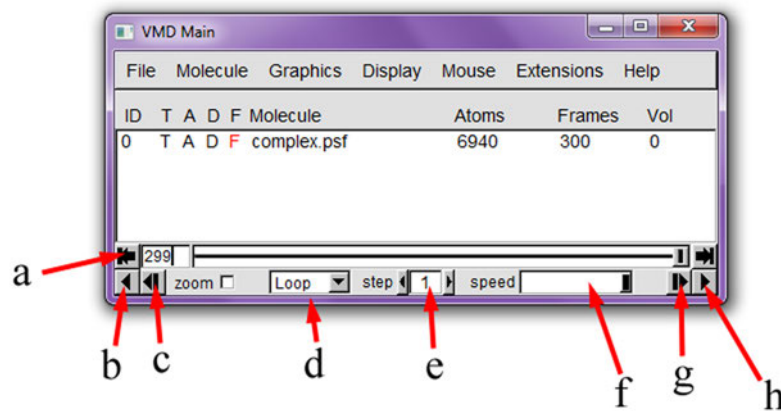







Figure 22: Animation Tools

- 1 In the VMD Main window, click on the  button in the lower left (Fig. 22a).
- 2 Try using the  button to jump to the end of the trajectory and go back to the beginning with the  button. You can see the final and initial

states of the trajectory, which correspond to the fully bound and partially unbound states of the complex.

You can click on the slider  and drag it back and forth to navigate through your trajectory. You can stop at anytime you want, or go at the speed you need. This is helpful when you are looking at a trajectory and want to spot the time when something interesting happens.

- 3 Using the slider, observe the behavior of calcium ions and the C-terminal β -strand hydrogen bonds throughout the trajectory. Partial unfolding occurs while the proteins unbind.




Unbinding versus Unfolding. The simulation example loaded in VMD was performed at 10 to 100 times faster pulling speeds than a loud sound-induced mechanical stimuli. During the trajectory, you can see partial unfolding of β -strands G in both Cdh23EC1+2 and Pcdh15 EC1+2, while the complex unbinds. At even faster stretching speeds, the complex unfolds without unbinding. At lower, closer to physiological pulling speeds, the simulations predict that the complex unbinds without any unfolding. Note that this simulation was performed with the complex in a water box, but coordinates for water molecules have not been included in the provided files to reduce their size.

On the lower part of the Animation tools (Fig. 22), you will find all the tools necessary to play an animation without using the slider. This is done with the Play buttons (b,c,g,h), that go forward and backward.

There are two ways to change the speed of your animation. You can adjust it using the Speed Slider (Fig. 22f). You can also adjust the step size. This is done using the Step Window (e). If this step is set to 3, the animation will show every 3rd frame, so it will make it faster.

- 4 Set the step to 5, and play the trajectory. Note that it plays faster, but it also looks less smooth than before. However, this can come handy if you are looking at long trajectories.

Looping styles. When playing animations, you can choose between 3 looping styles in the Style menu (Fig. 22d). These are “Once”, “Loop” and “Rock”. A nice trick:



- Go to the beginning of the trajectory (frame 0).
- Set the step number to the total number of frames in your trajectory (300).
- Set the style to “Rock”.
- Set the speed to lowest.
- Play (h).

You can see now a comparison between the first and the last frame of your trajectory. This trick is better than clicking on the Start and End buttons.

You can also jump to a frame in your trajectory, by entering the frame number in the window at the left of the Animation Tools.

NOTE: The Animation Tools you learned cycle through the frames of the Top molecule, but apply to all Active molecules.

3.3 Labels

VMD allows you to place labels to get information on a particular selection. We will now make use of those labels for fun and profit. Labels are selected with the mouse. In this example, we will cover labels that can be placed on atoms and bonds, although angle and dihedral labeling is also possible.

- 1 Choose the **Mouse → Labels → Atoms** menu item. The mouse is now set to “Display Label for Atom” mode. You can now click on any atom on your molecule and a label will be placed next to this atom. Clicking again on it will erase the label.

We will now try the same for bonds.

- 2 Choose the **Mouse → Label → Bonds** menu item. This selects the “Display Label for Bond” mode.

You will make a VDW Representation for the α carbon of each protein C-terminus, to be used for the bond label. In the pulling simulation, both are pulled in opposite directions at a constant velocity of 0.25 Å/ps.

- 3 Create a VDW Representation with the selection (**resid 208 and name CA and segname P1**) or (**resid 237 and name CA and segname P2**), and color it green (ColorID 7).

- Now that you can see the C-terminal C_α atoms, and the mouse is in the “Display Label for Bond” mode, click on both C_α atoms (one after the other). You should get a line connecting the two atoms. The number appearing next to the line is the distance between the two atoms in angstroms (Fig. 23).

Note that the color of labels can be changed using the Color Controls window available from the Graphics → Colors menu item (use the Labels category).

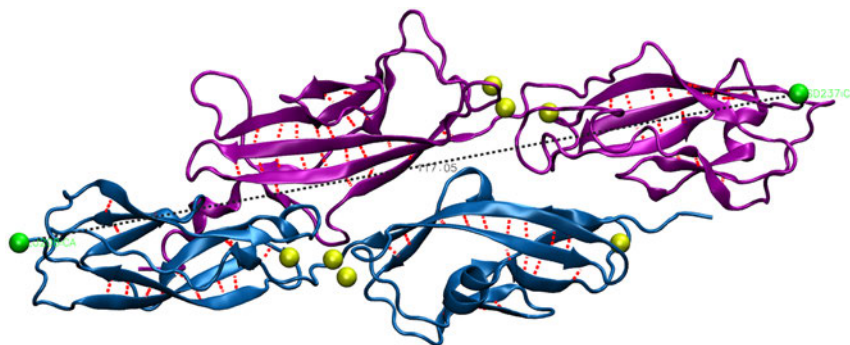


Figure 23: Bond selection of C_α terminal atoms in simulation. Both atoms selected display labels in green. The bond is shown in black (white in your screen), with the value of the distance between the atoms in Å displayed (117.05).

The value of the distance corresponds to the current frame.

There are more things you can do with labels using the Labels window.

- Select the Graphics → Labels menu item. The Labels window should pop-up.
- On the top left side of the window (Fig. 24), there is a pull-down menu where you can choose the type of label (Atoms, Bonds, Angles, Dihedrals). For now, keep it in Atoms. You can see the list of atoms for which you made a label.

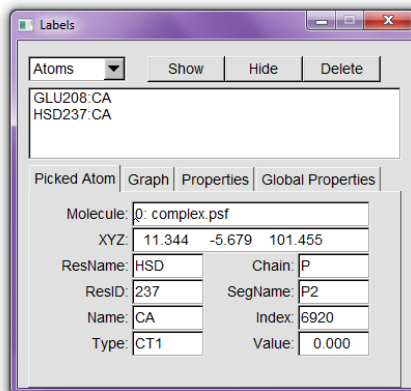


Figure 24: Label window.

- 7 Click on one of the atoms listed in the window. You can see all the information of the atom displayed. You can delete, hide, or show the label by clicking on the corresponding buttons.

The information about the atom corresponds to the current frame, and is updated as the frame is changed.

- 8 Now, in the Label window, choose the label type **Bonds**, and select the bond you labeled. Note that the information given corresponds to only the first atom in the bond, but the number in the Value field corresponds to the length of the bond in angstroms.
- 9 Select the bond you labeled between atoms **GLU208:CA** and **HSD237:CA**.
- 10 Click on the Graph tab and then click on the **Graph** button. This will create the plot of the distance between these two atoms over time (Fig. 25). You can also save this data to a file by clicking on the **Save** button, and then use an external plotting program to visualize the data.



Labels. The shortcut keys for labels are 1: Atoms and 2: Bonds. You can use these instead of the Mouse menu. Be sure the OpenGL Display window is active when using these shortcuts.

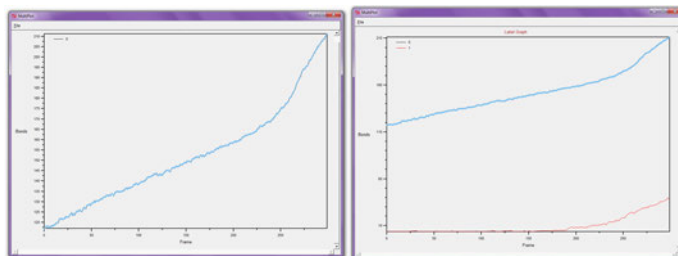


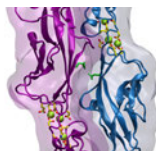
Figure 25: Plots of selected bonds over time created with the Graph button and VMD's internal MultiPlot program.

We will now create another representation that will help us monitor the separation of a salt bridge essential for the binding of Cdh23 EC1+2 to Pcdh15 EC1+2.

- 11 Click on the **Graphics → Representations...** menu item to open the Graphical Representations window. In the **Selected Atoms** text entry window, type **(segname P1 and resid 78)** or **(segname P2 and resid 113)**. Then select **Drawing Method → Licorice** and **Coloring Method → Name**. You should be able to see the salt bridge formed by Arginine 113 and Glutamate 78.

- 12 To create a bond, choose the **Mouse** → **Label** → **Bonds** menu item (or click 2 while the OpenGL Display window is active).
- 13 Click on the Arginine's C_ζ and Glutamate's C_δ atoms (one after the other). You should get a line connecting the two atoms of the side chains of these amino acids.
- 14 Select the **Graphics** → **Labels** menu item. The new bond should appear listed in the Label window. Select the new bond **GLU78:CD ARG113:CZ**, and while holding the “shift” key down on your keyboard, select the **GLU208:CA HSD237:CA** bond as well. Graph them by clicking on the **Graph** tab and then clicking on the **Graph** button.

The MultiPlot window should appear with two curves (blue and red) showing the length of the bonds throughout the loaded trajectory (Fig. 25). The time dependence of these bond lengths can help us distinguish unfolding from unbinding events.



Arginine 113 and Deafness. As you learned in previous parts of this tutorial, Cdh23 and Pcdh15 are essential for the vertebrate senses of hearing and balance. Human genetics has revealed that individuals carrying the R113G mutation (arginine replaced by glycine) are deaf (Zubair M. Ahmed et al, Human Genetics **124** 215, 2008). The structure of the Cdh23 and Pcdh15 complex suggests that replacing the long, charged arginine side chain for a neutral glycine weakens the interaction with its glutamate partner compromising the strength of the complex.

3.4 Tcl Scripting Basics and the Tk Console

VMD provides powerful tools for analysis of trajectories obtained using molecular dynamics simulations. These tools are best used through “scripts” that contain sets of commands to be executed to complete a task. VMD includes support for the Tcl/Tk scripting language. This section will attempt to provide the minimal amount of scripting knowledge that you will need to take full advantage of VMD.



The Tcl/Tk scripting language. Tcl is a rich language with many features and commands, in addition to the typical conditional and looping expressions. Tk is an extension to Tcl that permits the writing of graphical user interfaces with windows and buttons. More information and docs about the Tcl/Tk language can be found at <http://www.tcl.tk/doc>.

To execute Tcl commands, you will be using a convenient text console called Tk Console.

- 1 Select the Extensions → Tk Console menu item. A console window should appear with a prompt (Fig. 26). You can now start entering Tcl/Tk commands in it.

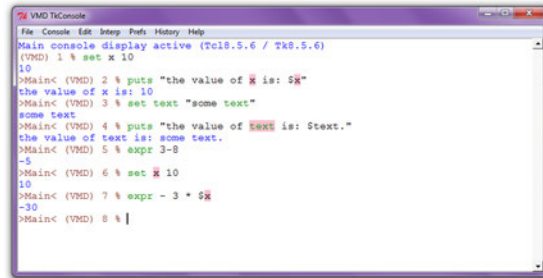


Figure 26: The Tk Console window.

You will initially focus on the very basics of Tcl/Tk. Here are Tcl's `set` and `puts` commands:

<pre> set <i>variable</i> <i>value</i> – sets the value of <i>variable</i> puts \$<i>variable</i> – prints out the value of <i>variable</i> </pre>

- 2 Try the following commands:

```

set x 10
puts "the value of x is: $x"
set text "some text"
puts "the value of text is: $text."

```

As you can see, `$variable` refers to the value stored in `variable`.

Here is a command that performs mathematical operations:

<pre> expr <i>expression</i> – evaluates a mathematical expression </pre>
--

- 3 Try experimenting with the `expr` command:

```

expr 3 - 8
set x 10
expr - 3 * $x

```

One of the most important aspects of Tcl is that you can embed Tcl commands into others by using brackets. A bracketed expression will automatically be substituted by the return value of the expression inside the brackets:

`[expr.]` – represents the result of the expression inside the brackets

- 4 Create some commands using brackets and test them. Here is an example:

```
set result [ expr -3 * $x ]
puts $result
```

3.5 The atomselect Command

You can edit atomic properties using VMD's `atomselect` command. The following examples will show you how. First, however, let's display our protein complex using a suitable representation.

- 1 Open the Labels window by selecting the Graphics → Labels menu item. Hide all atom and bond labels.
- 2 Open the Graphical Representations window using the Graphics → Representations... menu item.
- 3 Delete all representations using the Delete Rep button.
- 4 Click on the Create Rep button and type in `protein` as the atom selection. Change its Coloring Method to Beta and its Drawing Method to VDW. Your molecule should now appear as a white assembly of spheres.



The PDB B-factor field. The “B” field of a PDB file typically stores the “temperature factor” for a crystal structure and is read into VMD's “Beta” field. Since we are not currently interested in this information, we can recycle this field to store our own numerical values. VMD has a “Beta” coloring method, which colors atoms according to their B-factors. By replacing the Beta values for various atoms, you can control the color in which they are drawn. This is very useful when you want to show a property of the system that you have computed, which is not supported out-of-the-box by VMD.

You will now learn a very important Tcl command in VMD:

`atomselect molid selection` – creates a new atom selection

The first argument to `atomselect` is the molecule ID (shown to the very left of the VMD Main window), the second argument is a textual atom selection like what you have been using to describe graphical representations in unit 1. The selection returned by `atomselect` is itself a command which you will learn to use.

- 5 Type `set p [atomselect top "protein"]` in the Tk Console window. This creates a selection containing all the atoms in the molecule and assigns it to the variable `p`. Instead of a molecule ID (which is a number), we have used the shortcut “`top`” to refer to the top molecule.

The result of `atomselect` is a function. Thus, `$p` is now a function that performs actions on the contents of the “protein” selection.

- 6 Type `$p num`. Passing `num` to an atom selection returns the number of atoms in that selection. Check that this number matches the number of atoms for that molecule (as read from the VMD Main window), except for the 7 calcium ions.
- 7 Type `$p set beta 0`. This resets the “beta” field (which is displayed) to be zero for all atoms.

Atom selections reference the atoms in the original molecule. When you change a property (*e.g.* beta value) of some atoms through a selection, that change is reflected in all the other selections that contain those atoms.

- 8 Now, type `set sel [atomselect top "hydrophobic"]`. This creates a selection containing all the hydrophobic residues.
- 9 Let's tag all hydrophobic atoms by setting their beta values to 1. You should know how to do this now: `$sel set beta 1`. If the colors in the OpenGL Display do not get updated, click on the **Apply** button at the bottom of the Graphical Representations window.



Examples of atomic properties. You can get and set many atomic properties using atom selections, including segment, residue and atom names, position (x, y and z), charge, mass, occupancy and radius, just to name a few.

- 10 You will now change a physical property of the atoms to further illustrate the distribution of hydrophobic residues. Type `$p set radius 0.5` to make all the atoms smaller and easier to see through, and then `$sel set radius 1.5` to make the hydrophobic residues larger. The radius field affects the way that some representations (*e.g.*, VDW, CPK) are drawn.

You have now created a visual state that clearly distinguishes which parts of the protein are hydrophobic and which are hydrophilic. If you have followed the instructions correctly, your protein should resemble Fig. 27.

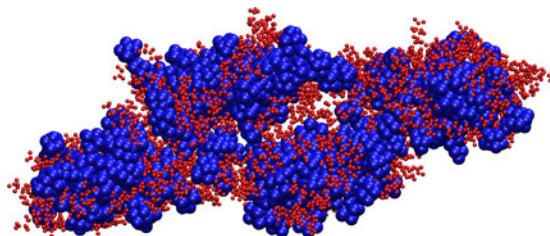
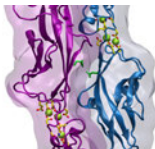


Figure 27: Tip link Cdh23 and Pcdh15 handshake in VDW representation, colored according to the hydrophobicity of its residues.



Hydrophobic residues. As you probably noticed in your rendering of the Cdh23 EC1+2 and Pcdh15 EC1+2 complex, hydrophobic residues are mostly contained in the inner core of each of the four EC repeats. This is typical for small water-soluble proteins. As the protein repeats fold, the hydrophilic residues will have a tendency to stay at the water interface, while the hydrophobic residues are pushed together and play a structural role. This helps the protein repeats achieve proper folding and increases their stability.

Atom selections are useful not only for setting atomic data, but also for getting information. Let's say that you wish to communicate which residues are hydrophobic, all you need to do is to create a hydrophobic selection and use `get` instead of `set`.

- 11 Use `get` with your hydrophobic `sel` atom selection:

```
$sel get resname
```

But there is a problem! Each residue contains many atoms, resulting in multiple repeated entries. Can you think of a way to circumvent this? We know that each amino acid has the same backbone atoms. If you pick only one of these atoms per residue, each residue will be present only once in your selection.

- 12 Let's try this solution. Each residue has one and only one α -carbon (name CA = alpha):

```
set sel [atomselect top "hydrophobic and alpha"]
$sel get resname
```

Aha! It worked.

- 13 You can also get multiple properties simultaneously. Try the following:

```
$sel get resid
$sel get {resname resid}
$sel get {x y z}
```

- 14 Once you are done with a selection, it is always convenient to delete them in order to save memory by typing:

```
$sel delete
```

3.6 An Example Tcl Script: Molecular Separation in a Trajectory

VMD is a powerful tool for MD analysis. In this section you will use Tcl scripts to compute the distance between the center of mass of Cdh23 and Pcdh15 throughout our unbinding trajectory.

- 1 The script we are going to use is called `distance.tcl`. This is the script content:

```
set outfile [open distance.dat w]
set nf [molinfo top get numframes]
set mol1 [atomselect top "segname P1"]
set mol2 [atomselect top "segname P2"]
# distance calculation loop
for { set i 1 } { $i <= $nf } { incr i } {
  $mol1 frame $i
  $mol2 frame $i
  set cmmol1 [measure center $mol1 weight mass]
  set cmmol2 [measure center $mol2 weight mass]
  set distance [veclength [veclength [vecsub $cmmol1 $cmmol2]]]
  puts $outfile "$i, $distance"
}
close $outfile
```

- 2 The script does the following:

- Opens file `distance.dat` for writing


```
set outfile [open distance.dat w]
```
- Gets the number of frames in the trajectory and assign this value to the variable `nf`

```
set nf [molinfo top get numframes]
```
- Makes a selection with atoms of Cdh23.


```
set all [atomselect top "segname P1"]
```
- Makes a selection with atoms of Pcdh15.


```
set all [atomselect top "segname P2"]
```

- Text after # denotes comment.

```
# distance calculation loop
```
- Loops over all frames in the trajectory:

```
for { set i 1 } { $i <= $nf } { incr i } {
```
- Changes the selection \$mol1 to the frame being used for the calculation (note that the content of the selection must be identical in every frame, otherwise a \$sel update command needs to be added).

```
$mol1 frame $i
```
- Changes the selection \$mol2 to the current frame.

```
$mol2 frame $i
```
- Calculates the center of mass of the atoms stored in \$mol1 and assigns it to cmmol1 as a vector.

```
set cmmol1 [measure center $mol1 weight mass]
```
- Calculates the center of mass of the atoms stored in \$mol2 and assigns it to cmmol2 as a vector.

```
set cmmol2 [measure center $mol2 weight mass]
```
- Subtracts the vector stored in \$cmmol2 from \$cmmol1, then computes the length of the vector resulting from the subtraction and assigns it to distance.

```
set distance [veclength [vecsub $cmmol1 $cmmol2]]
```
- Writes the calculated distance to file:

```
puts $outfile "$i, $distance"
```



Atom selections inside loops. The `distance.tcl` script described above does not create selections inside the distance calculation loop. However, if you need to create selections using the `atomselect` command inside a loop, always remember to delete them using the `$sel delete` command. Otherwise you will run out of memory.

Now that you understand the script, you can use it for the system loaded in VMD to compute the distance between the center of mass of Cdh23 EC1+2 and Pcdh15 EC1+2.

- 3 Type `source distance.tcl` in the Tk Console (be sure that your current directory is `vmd-tutorial-files`). This will perform all the commands in the script. The script will write a file `distance.dat` that will contain the value of the separation between the two molecules as a function of frame number.

Outside of VMD, you can use some plotting program to see this data. Examples of these are gnuplot, xmgrace, excel, Mathematica (Fig. 28).

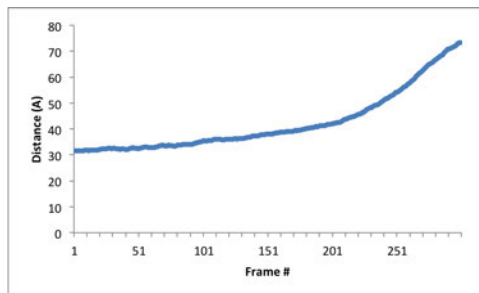


Figure 28: Plot of distance between the centers of mass of Cdh23 and Pcdh15 as a function of frame number.

This ends our “Hands-on Introduction to Protein Visualization” tutorial. We hope that you learned a lot with it, and that you will make great use of all the capabilities VMD has to offer. Please visit <http://www.ks.uiuc.edu/Training/Tutorials/> and <https://research.chemistry.ohio-state.edu/sotomayor/teaching/> for updated versions of this and other tutorials.