ePMV: INTERFACE GLOSSARY July 2011 AMI Workshop

Written by Graham Johnson and Fabian de Kok-Mercado

Made possible by **Ludovic Autin**, Graham Johnson, David Goodsell, Michel Sanner & Arthur Olson

http://epmv.scripps.edu **Download free, install, tutorials, gallery, forum**

Get a Molecule:

1. Browse: Use to load ProteinDataBank (PDB) files from your computer

2. Fetch: Directly download PDB file variants with 4-character code and an internet connection

a. PDB: Protein Data Bank, *pdb.org*. This archive contains 3D coordinates and detailed information about the structures of molecules (proteins, nucleic acids, and macromolecular assemblies) that have been experimentally determined by researchers using x-ray crystallography, Nuclear Magnetic Resonance, homology modeling and hybrid methods.

b. TMPDB: Trans-Membrane Protein DataBank, *tmpdb.hu*, files from the PDB that are computationally determined by the website's algorithm to be membrane proteins. A TMPDB protein's transmembrane region is centered at the origin and pointed outward along -X, thus the YZplane connotes the center of the lipid bilayer and "outward" depends on the compartment, but always points from the deeper towards more superficial membrane-bound compartments to the left.

c. OPM: Orientations of Proteins in Membranes database, *opm.org*. Similar to the TMPDB, OPM takes transmembrane orientation one step further by performing a TM molecular dynamics relaxation on the molecule before posting it into the database. An OPM protein's transmembrane region is centered at the origin and pointed outward along +X, thus the XY plane connotes the center of the lipid bilayer. These files also come with the hydrophobic boundaries of the particular membrane shown as red (outer) or blue (inner) spheres. Not all membrane-associated proteins fully cross the membrane (integral) so only one membrane indicator may be shown for non-integral transmembrane PDBs. (red = superficial, blue = deep)

d. CIF: Crystallography Open Database, aims to gather all available inorganic, metal-organic, and small organic molecule structural data.

e. PQS: Protein Quaternary Structure Query, makes coordinates available for likely quaternary states for structures contained in the PDB.

Selections:

1. Current selection: pulldown list of available selections. Imported molecules are available by default. A user can access saved selections of any type here as well. The state of the molecule (or selection) will update the rest of the GUI to correspond. GUI actions only work on the current selection!

a. Save as PDB: Save the current selection as a PDB file to your harddrive.

2. Add selection set using string or <keywords> : subselections of the current selection can be made from the pulldown list of provided keywords or by typing PMV selection syntax into the box. Sophisticated selections can be made quickly and easily once familiar– asyntax glossary is available at: ePMV.scripps.edu/documentation/epmv-overview click on the *instructions* link under Advanced Features The overriding hierarchy is Molecule:Chain:Residue:Atom, for example, 1XYZ:B:ALA:CA, will select all of the α-carbons (CA) of all of the Alanines (ALA), in chain B, of loaded molecule 1XYZ. Keywords provide common yet powerful shortcuts that can be used alone or incorporated into more complex selections.

a. Picked: A special keyword worth noting. *Picked* scripts a selection set from atoms selected in the host hierarchy (ObjectManager in C4D)

3. Save set: The selection string and state of the GUI that is operating on the current selection can be saved and later accessed under the Current Selection pulldown.

a. Rename set: *Rename a* selection set *that was saved with* Save Set*.*

b. Delete set: Delete a selection set that was saved with *Save Set*. The actual atoms won't be deleted, but the selection set will no longer be available under *Current Selection*. Helps to cleanup the pulldown menu.

4. Delete atoms in the selection set: Always delete ePMV objects using this button command–deleting from the C4d object manager will corrupt your current ePMV project and functionality will be lost. The atoms in the selection set will be deleted from the ePMV object. Objects like MSMsurfaces will need to be rebuilt to see the effect, but atoms will be updated immediately.

Preset Representations:

1. Lines: Redundant menu item runs a script that checks the *Lines* radio button with the same result.

2. Licorice: Runs a script that checks the *Sticks* radio button and sets the *Stick Ratio* to 1 to create thick/rounded lines.

3. SpaceFilling: Runs a script that checks the *Atoms* radio button.

4. Ball+Sticks: Small atoms are connected with covalent bonds represented as cylinders according to the proper chemistry. The Scale and Ratio can be adjusted with the sliders.

5. RibbonProtein+StickLigand: One step button runs a script that creates a cartoon ribbon for proteins and nucleic acids and a stick representation for ligands.

6. RibbonProtein+CPKLigand: One step button runs a script that creates a cartoon ribbon for proteins and nucleic acids and a spacefilling representation for ligands.

7. Custom: Saves the state of the current selection as a representation style.

8. Save Custom As: Set the name for the custom representation.

Atom/Bond Representations:

1. Atoms: (CPK) Each atomj of a PDB file is represented by a sphere that approximates the limits of its electron density or Van der Waals radii. This *spacefilling* representation gives viewers a better feeling of shape and space occupied by the overall molecule, but in general is too visually complex to effectively represent large molecules or assemblies. Colored by CPK (canonical atom colors) by default

a. Scale: Adjusts relative radii of all atoms in the selected chain by scaling the base atom from default of 1.

2. Sticks: Small atoms are connected with covalent bonds represented as cylinders according to the proper chemistry. The *Scale* and *Ratio* can be adjusted with the sliders. A *Ratio* > 1 creates a *Ball-n-Stick* model. Colored by CPK (by atom type) by default.

a. Scale: Adjusts the relative radii of all atoms and bonds in the selected chain.

b. Ratio: Adjusts size ratio as baseAtomRadii/bondRadii

3. PointClouds: Each atom is represented as a point in space. Displays where each atom lies but this shorthand representation shows no further chemical information. The coordinates can be seen under the C4D *Structure menu*. You may note here that the X and Z coordinates are transposed from the PDB file for each atom– this is because the PDB coordinates are for a right-handed Cartesian coordinate system… Cinema 4D uses a left-handed system.

4. Lines: Each atom is a point in space connected by covalent bonds represented as lines. This is the default representation of most molecular viewers. Useful for efficiently visualizing detailed chemistry at close range. Rarely used in publication.

Backbone Representations:

Backbone representations connect the α-carbons of each amino acid with various geometry types– generally different cross-sections swept along a 1.5D spline, where each anchor connects the α-carbons.

1. Ribbons (Cartoon): Stylized representation of a backbone model where peptide regions that form α-helices and β-sheets are widened into a stylized form. This schematic illustrates the topology of the chain of proteins and nucleic acids and is essential shorthand for communicating the folding of biomolecules. Developed by Irving Geis and formalized by Jane Richardson.

2. BeadedRibbons: A fully customizable *ribbon cartoon* reminiscent of the *Fancy Ribbon* style developed by MolScript in the 1990s and popularized by PyMOL… with lots of bells and whistles

a. Options: allows user to customize dozens of parameters in a large palette to create unique styles that retain correct backbone geometry.

3. Armature: *IK Joint* skeletons that can be rigged/bound to model. Use with caution as this feature makes is very easy to create physically irrelevant structures that will make content experts cringe.

4. Loft (Will Be renamed Sweep): builds a backbone spline into a Sweep object to create a *backbone worm* representation. The circle may be replaced with any cross-section spline object. Easily animateable. ***4.5 Loft will return in future editions… temporarily on hold:*** Builds a series of cross-section instances at each alpha carbon from native host splines into a loft object. Will allow for fully customizable cross-sections for helices, beta strands, turns and coils as well as smoothly animated transition states.

5. Spline: Builds a spline with an anchor point at each a α-carbon. A temporary bug requires you to switch from Cubic to Linear, then back to Cubic in the Attributes Manager to get a smooth spline.

Surface Representations:

1. MSMSurf: Michel Sanner Molecular Surface. Imagine a rubber skin wrapped around an *Atoms* representation. More accurately, a *probe sphere* is rolled around the entire molecule and the MSMsurface is a resulting mesh that represents the probe's closest approximation.

a. probe radius: The lower the radius, the tighter the skin is wrapped between the atoms because the probe can "squeeze" into tighter cavities. A radius of 0.001 can create a single mesh that approximates a CPK model.

b. triangle density: Subdivides surface (adds or removes triangles)

2. CoarseMolSurf: Creates "blobby" models of molecules that are more appropriate for complex mesoscale structures and assemblies such as autoFill models.

a. isovalue: Defines the contouring threshold

b. resolution: Determines the radius of the skin in relation to the atom surfaces

c. grid size: Determines the mesh detail (number of polygons in the mesh polyhedron)

3. Metaballs: Builds an atomic point cloud into a metaball object to allow 3D animation users to build and adjust a crude surface with more familiar settings. Future versions will have color options. For now it can be colored using the APBS approach on the website or by splitting clouds into atom types and using proximal shaders with appropriate falloffs. Currently the best option for smoothly animating surface representation conformation changes.

Color By:

Some color types don't affect certain representations. Ribbons are not yet fully colorable by gradient options– each secondary structure polyhedron will receive the color of the first amino acid in that region. Many of the default colors can be altered via the color-coding palette– Edit>ColorPalettes. After changing a color in the palette, the ColorBy: dropdown must be reapplied to glue it to the current selection

1. Atoms Using CPK: Traditional color scheme according to atom (element) by Corey, Pauling, and Kulton (carbon=black, oxygen=red, nitrogen=blue, hydrogen=white, sulfur=yellow). See options>color palette.

2. Atoms DG (polarity/charge): Colors hydrogen atoms the same as the atom to which they are bonded. It also changes carbon to white, and uses pastel shades for oxygen and nitrogen that do not carry a formal charge. Fully saturated reds carry a negative charge and fully saturated blues carry a positive charge. It highlights the shape of the molecule and also reveals carbon-rich amino acids that interact with the cellular membrane. Go under options>color palette to see color coding.

3. Per residue: Amino Acids or Nucleotides color coding scheme. Change defaults with color-coding palette– Edit>ColorPalettes. This is something PyMol requires a plugin for.

4. Per residue shapely: Residues color scheme derived from Rasmol.

5. Secondary structure: Color codes amino acids as belonging to α-helices (default is pink), β-strands/sheets (yellow), random coils (white), and Turns (blue). Change defaults with the color-coding palette– Edit>ColorPalettes. To color atoms/bonds/surfaces by secondary structure, you must first build a standard *ribbon* representation to the scene (it need not be visible, you can just build it once then uncheck the box.)

6. Chains: Colors each amino acid chain differently. Change defaults with color-coding palette– Edit>ColorPalettes. Works on all representations except line and point.

7. Custom color: Click the *color box* to define a color then apply it to your *current selection*.

8. Rainbow for N to C: Rainbow gradient from the N-Terminus (translation start of protein) to the C-Terminus (end of protein/amino acid chain). ROYGBIV in reverse (N=blue, C=Red) Communicates tertiary structure associations and domain localization of relative regions of the linear peptide sequence at a glance.

9. Temperature Factor: Spectral color scheme based on protein anisotropic temperature (B factor). Used to describe attenuation of x-ray scattering or coherent neutron scattering caused by thermal motion. (blue=cold, red=hot)

10. sas area:

UV Texture mapping:

1. Create Texture Mapping for: Type the name of the ePMV molecular object (name found in the ObjectManager hierarchy), that you'd like to texture. Must be a polygonal mesh object.

2. Using: Select the method you'd like to use to unwrap the mesh.

a. unwrapped mesh UV:

b. regular disposed triangle:

3. Create: Type the path to your current C4D file and give the material a name, then click [Create]

Be Patient, large meshes can take several minutes.

Data Player:

1. Overview: The *Data Player* window is a context-sensitive scroll bar that allows you to do things like efficiently play through multiple 3D conformations of a protein as if in a movie, regardless of the representation style, or to adjust relevant parameters, like the isosurface value of a volumetric 3D map file to create a contour mesh.

Files that have multiple states of a PDB built into them with the *MODEL #* syntax, e.g. NMR files, that get loaded into *Get Molecule*, will automatically appear in the *Data Player* pulldown to allow a user to scroll through all of the conformations.

Other files that refer to a compiled data trajectory file, for example, require you to load one state of the molecule as a PDB file, then to load a trajectory file second. The PDB and the trajectory file must have the exact same number of atoms in the exact same order and with the exact same atom names- to ensure this, you should save a single PDB state from your MD program or viewer at the same time you export your .xtc or .trj file. For example, on our tutorial page, you can Get Molecule 1bta.pdb to build any representation of the protein, and load 1bta.xtc in the Data Player to scroll through 1000 conformations of the protein that resulted from a molecular dynamics simulation performed by Ludovic Autin using GROMACs. We recommend using the lines representation for fast scrolling, but they all work (be careful scrolling with a surface representation.) You need to rebuild ribbons after you stop scrolling to see the updated conformation.

As of July 2011, you must first load any PDB file before you can load a 3D map file to trigger the necessary libraries into action.

2. Browse: Browse your harddrive(s) to load a data file of one of the types listed under the <e.g.> pulldown.

a. e.g.: A list of the different file types you can load (not all are perfectly functional as of July 2011 beta)

3. Apply: Applies the *step value* associated with the trajectory or map file to the data selected in this pulldown, e.g., you can have more than one trajectory file loaded for a molecule but the one selected here will be applied to the current selection (assuming the atoms matched as described in the *Data Player Overview* above.

4. Step/Value: A context sensitive value displayed and adjustable here will alter a relevant parameter specific to your filetype as described in the *Data Player Overview* above. It will automatically adjust to display a relevant range (e.g. .map files have negative reals and .trj files have only as many positive integers as are in the file.)

PMV-Python scripts/commands:

1. Open: Open an existing python command that comes installed with ePMV

2. Save: Modify an existing command or type your own and save it to disk here.

3. Command line: Type Python commands you want to execute here. This command box is the same as the default command line, but offers more space and syntax color for easier coding.

4. Exec: Click to execute the python script visible in the Command Box.

ePMV: TOP DROPDOWN MENU GLOSSARY

Written by Graham Johnson and Fabian de Kok-Mercado

Made possible by **Ludovic Autin**, Graham Johnson, David Goodsell, Michel Sanner & Arthur Olson

File

1. ***Recent Files:*** Lists recently opened files for easy access
2. ***Open PDB:*** Same as the *Get Molecule* [Browse] button, allows user to open files saved on computer
3. ***Save PDB:*** Allows user to save a pdb file modified in the host, does not save ePMV settings, only atomic coordinates. To save the ePMV settings, just save your host file using File>SaveAs in the host GUI menu. In August 2011, you'll be able to rebuild the exact connection and ePMV GUI state of your host molecules.
4. ***Open Data:*** Same as the *Data Player* [Browse] button

Edit

1. Options

a. Center Molecule: Translates the atoms from the crystallographic coordinates to position the molecule's center of mass at the origin of the host scene upon loading

b. Femove Water: Deletes waters from the ePMV instance of the PDB file upon loading.

c. Synchro realtime: Enables changes that a user or function makes to atom positions in the host viewport to alter the underlying ePMV instance of he molecule. If checked, saving a PDB file after moving atomic spheres or points will generate a modified PDB file with coordinates that reflect the changes. Simulators like Extensions>Modeller will allow for realtime molecular dynamics in the viewport following user interaction.

d. PMV camera: Adds a host camera to the scene with settings that exactly match the default camera in the PMV viewport exactly. (PMV is the standalone molecular viewer that underlies ePMV.)

e. Use progress bar: Show a progress bar in the host's native GUI when an ePMV process takes more than a few seconds to let you know that ePMV is still calculating something challenging.

f. Split bonds: Generate two bond instances at each bond (one from each covalent participant) instead of one bond instance that has a gradient color. By default, this would create a grey and a red cylinder meeting in the middle of a carboxyl bond which could add geometry and cause issues when rendering stick models with transparency.

g. Force downloading on Fetch: If you fetch different instances of molecules with the same name as one that still exists in the PMVdownloadedMoleculeCache, the cached version of the molecule will be used. This is faster but causes an issue if you want to, for example, load the PDB, TMPDB, and OPM version of a transmembrane protein (they have the same name from each database e.g., 1xyz.pdb) into the same scene to compare the different files. Unless this box is checked, the version of the molecule first fetched would be used each time. Forcing download will import the molecule desired from the correct database regardless of the molecules name at the database or in cache.

h. PMV light: Adds a host light (or sun) to the scene with settings that exactly match the default light in the PMV viewport exactly.

i. Use Modeller: When checked, if a PDB file is loaded from the harddrive (won't work if Fetched), a new Modeller-specific version of the PDB file will be generated in the same directory with cleanups needed by Modeller to perform MD simulations. For example, with *Use Modeller* checked, ePMV will save 1crn.pdb, when loaded to a new file 1CRNm.pdb in the same directory and 1CRNm.pdb will be loaded instead of 1crn.pdb as indicated in the Current Selection pulldown name..

j. Use PyMol: Allows PyMOL geometry to be loaded into ePMV directly, e.g., (Molscript style beaded ribbons, Caver, relaxed models, Ligand functional group *Building*, etc.)

k. Center Grid: Center 3D grids that correspond to molecules or volumetric maps- similar to Center Molecule as described above.

l. Synchronize data player to time: When repaired in August, this will allow you to scroll through the current data player using the host timeline controls to easily set up keyframes, etc.

1. **Steps every:** Take this many time steps…
2. **Frames:** … per this many host-timeline frames.

2. Color palettes

a. atoms type: refers to *Atoms Using CPK*

b. atoms polarity type: refers to *Atoms DG (polarity/charge)*

c. residue type: refers to *Per residue*

d. residue shape type: refers to *Per residue shapely*

e. secondary structure: refers to *Secondary structure*

f. chains: refers to *Chains*

g. reset to: reset all colors to the PMV defaults

h. set color: apply the colors you altered to the vector based and independent materials

3. Delete water: Remove all water molecules from the ePMV instance of the PDB file

4. Hydrogens:

a. delete hydrogens: Remove all hydrogens from the ePMV instance of the PDB file

b. add hydrogens: Build hydrogens onto the current selection obeying rules of chemistry.

5. Biological unit: Many molecular structures exist in quaternary conformations in cells that are different from the quaternary conformations revealed in the crystallized state. The default download for 2plv.pdb (polio virus capsid protein, for example, does not look like a spherical (icosohedral) capsid that you see representing the infamous biological state of this molecule. The protein data bank provides two mechanisms that allow molecular viewer software to show biologically relevant version. On the PDB website, you have the option of downloading the Asymmetric unit or the biological unit(s). The biological unit and asymmetric unit are often the same, for example, in 1crn.pdb (you can toggle back and forth between static representations of them in the PDB image view at pdb.org.) You have the option of downloading the biological unit (sometimes there is more, which in the case of 2plv.pdb will be a file 60x larger than the asymmetric unit. The other choice is to download a properly formatted, modern version of the asymmetric unit and to construct the other biological units from a list of matrix transformations found in the newer versions of the files (REMARK 350 lines). Edit>Biological Unit does this for you.

a. Edit>Options: Uncheck Center Molecule

b. Edit>Biological Unit: Copy and paste the name of the ePMV representation that you want to instance into the box, hit run, and wait. If there are BIOMT coordinates in the REMARK 350 lines of the file, a new parent will appear with the appropriate number of instances below. Some molecules will have only one representation that fits over the original exactly (i.e., the asymmetric unit = the biological unit), Some will have a simpler molecule (i.e., the biological unit is smaller than the asymmetric unit), and some will be larger, a dimer, a trimer, a multimer, with certain viruses having several hundred copies of the asymmetric unit in the Biological unit.

6. Apply transformation: PDB model elements moved by the user (such as changing the position of an atom(s), will be applied to the underlying PMV model.

a. From: Picks which modified elements to send back to the PMV model.

1. Cpk: Send coordinate changes of Atom from viewport to the model

2. Lines: Send coordinate changes of PointCloud under lines from viewport to the model

3. Bones: Send coordinate changes of bones to associated atoms to the model

4. Splines: Send coordinate changes of viewport splines to associated atoms

Extensions

Up to date descriptions of third party modules that extend ePMV functionality and installation instructions can be found under the Extensions button on the wiki at http://ePMV.scripps.edu Although the following menu items exist in the default install, some of them require the installation of additional python software as instructed on the website.

1. APBS Electrostatics (plugin): Allows any geometry with a UV coordinate tag to be colored by an electrostatic potential map that is either imported or generated directly from a model in the scene. The user can, for example, import a molecule, generate an APBS color grid based on the atoms in that model, generate a host sphere, convert the sphere to polygons, and map the intersecting colors from the grid onto the sphere

a. readgrid

b. readpqr

c. run

d. normal offset

e. std scale

        f. geometry color

2. PyAutoDock
    a. receptor
    b. ligand
    c. type of score
        1. PyPairWise
        2. Ad3score
        3. Ad4score
    d. Display label
    e. Real time
    f. Color Rec
    g. Store
    h. Color LIg
3. Add an Extension

Help

1. About ePMV: useful to see the current version and contributing authors and any licensing or copyright needs as they come up.

2. ePMV documentation: links to the current user manual and reference files on the epmv.scripps.edu wiki website

3. Check for Update: As of version 0.3.2, this button will offer you a chance to automatically update your ePMV software. It provides the option to archive your current version in case you prefer to revert by hand. You can opt to keep up with the latest experimental build that may change daily, or to retain the stable version of the software until the next stable release is avaialbe- either way you can update in one click with this menu item.

4. Citation Information: Several contributing algorithms need to be cited if you plan to publish a modified version of the software or refer to it in a paper. We ask that images made with ePMV, especially journal covers, additionally cite ePMV- for example, *Cover image created by John Doe with ePMV…*