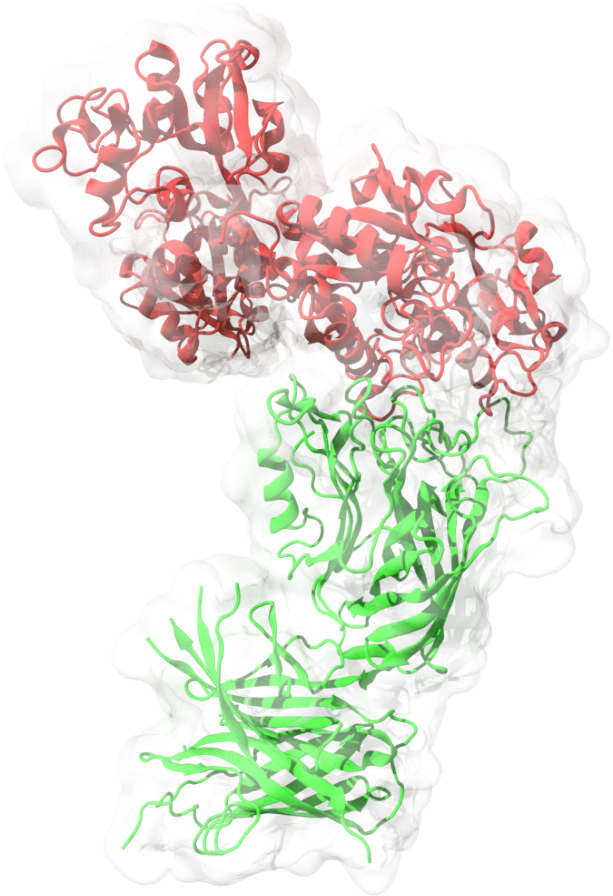
Making Beautiful Figures

Nicholas K.H. Ostan



Contents

1 Quick Introduction

I want to share with you some of the ways I have learned to make beautiful figures. I think scientific communication is more effective with simple, appealing, straightforward figures. Stop putting PowerPoint garbage into your presentations, and make a real impression on your audience. After all, this is your *hard work* that you are displaying! If you do end up making nice figures, just please don't submit to the same journals as me.

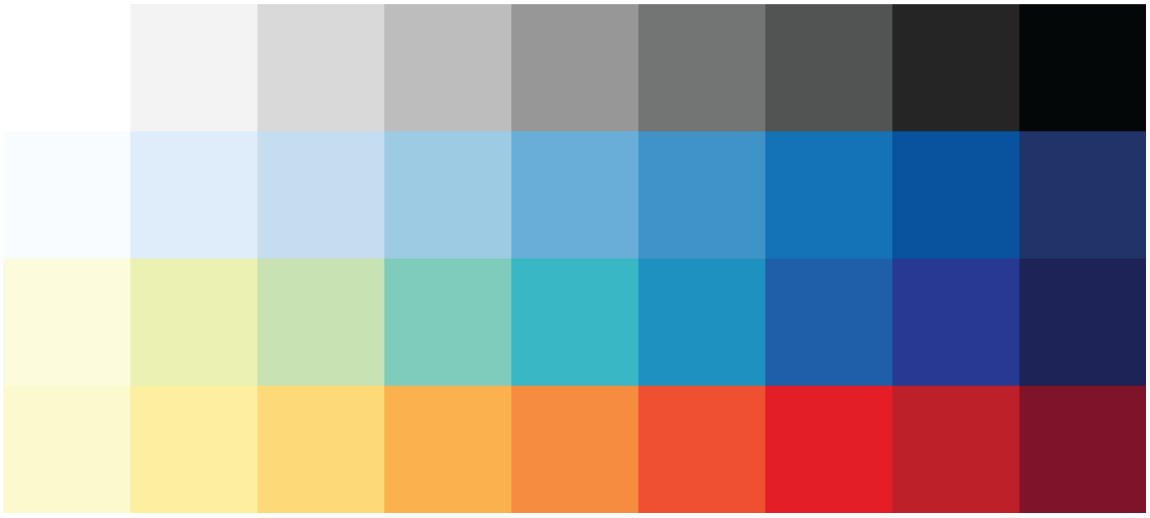
My goal here is to work through ∼1 page length examples with broad applicability so that you can modify things to your own taste. This is by no means a comprehensive resource on the subject. It can be very hard to navigate the wiki pages and user manuals for all of the presented software, so consider this handbook the **fun**damentals. You should have some low level working-knowledge of how to operate a mouse and keyboard. You should have also played around in Adobe Illustrator a little bit beforehand. I will use several software packages in the following examples. Feel free to skip past any parts/ you don't care to learn about. The software I will cover includes:

1. **Adobe Illustrator**
2. **PyMOL**
3. **VMD**
4. **Blender**

If you are submitting your research to some journal, you must follow the guidelines of the journal at the end of the day. Anyways, I want to keep this short. I hope you learn something from this little handbook!

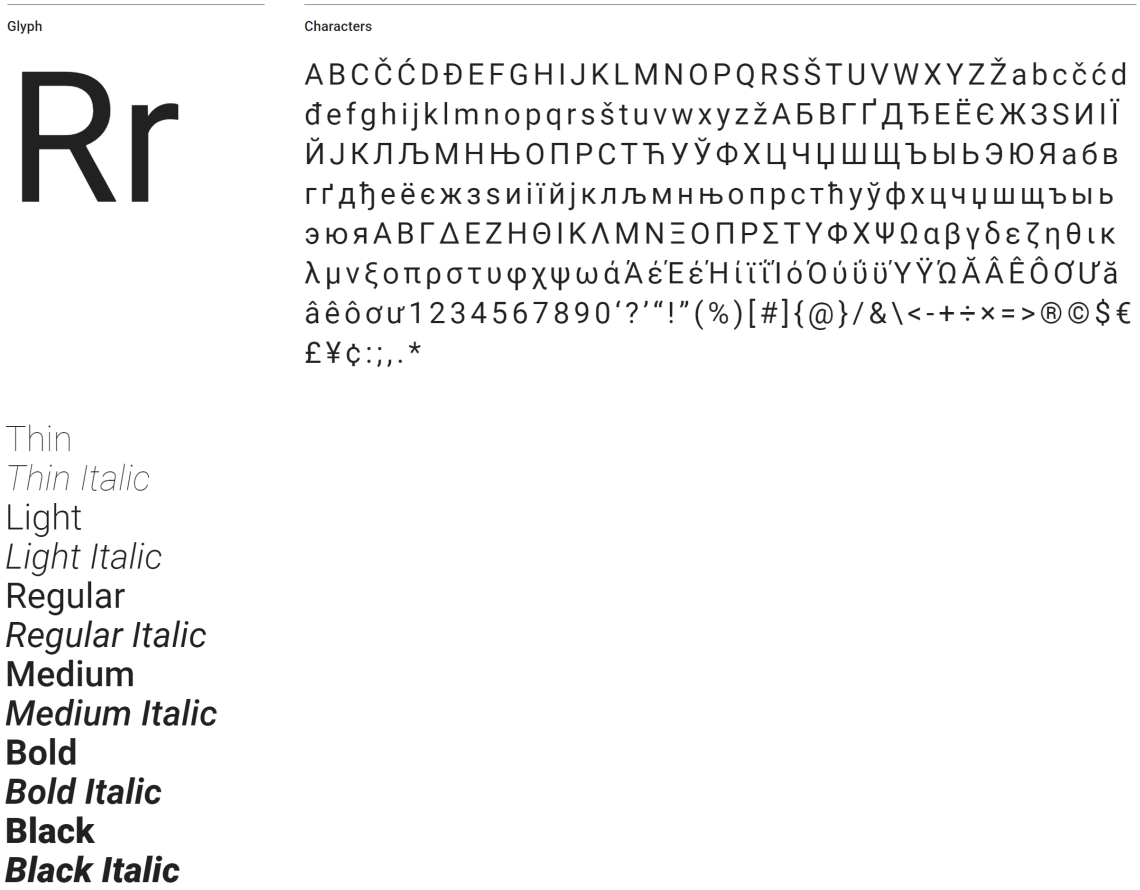
2 Colour Schemes

You may be surprised to know that there is an entire theory devoted to the mixing of colours called **colour theory**. There are a few things to consider when picking colours for scientific pieces. Firstly, 1 in 12 men are colour blind (1 in 200 for women)! That means it is important to avoid certain *combinations* of colors. These include green & red, green & brown, blue & purple, green & blue, and blue & grey. To account for this, it can be useful to integrate textures (such as checkerboxes, cross-hatching, etc.) into your figures where possible. If you want your figures to look professional, find a color scheme, such as this one, and bring it into your Adobe Illustrator or Adobe Photoshop file. Use the eyedropper to pick from the desired color so that you stay **consistent**.



3 Fonts

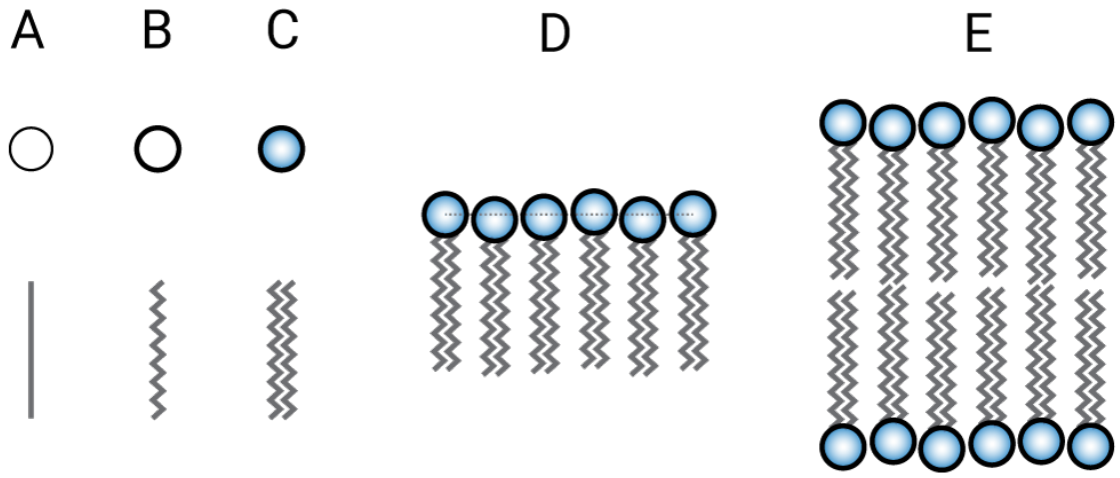
Many journals request that figure fonts be of Arial typeface. The font Roboto is in the same font family as Arial, and is slightly more polished. Roboto can be downloaded for free from Google fonts, at this link: **https://fonts.google.com/specimen/Roboto**



Use the same font in all of your figures, unless of course you require a monospace font for sequence alignments. There are plenty of professional monospace fonts on Google fonts as well.

4 Drawing Cell Membranes

Drawing cell membranes is an awesome thing to do for people reading your work, where relevant. If you are trying to display some interaction, the membrane acts as an 'anchor' point for the reader or viewer to orient themselves. The conceptual picture you are trying to display is much clearer when the viewer can picture not only what is happening, but also where it is happening. Below is a general breakdown of how you can make a nice lipid bilayer. Let's go through each step together:

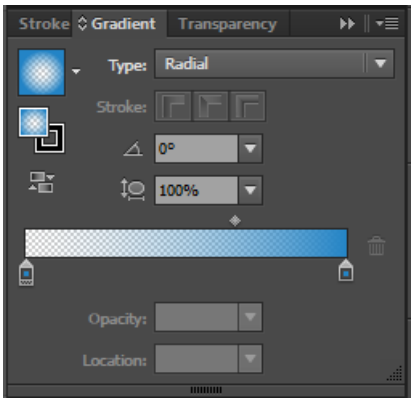


(A) Start by drawing a circle of 1pt stroke with white fill. Additionally, draw a 2pt grey vertical line.

(B) Change the circle stroke thickness to whatever you would like. Here I am using 2px. Select your vertical line, and go to **Effect** > **Distort & Transform** > **Zig-Zag**. Select the 'Preview' checkbox at the bottom left of the window so you can see changes in realtime. Modify the ridges per segment to 14, and the depth to 2. You can choose smooth or round depending on how you would like your

membrane depicted. Note this is also a fantastic tool for drawing sine/cosine waves. Overlaying two rounded zig-zag segments makes a nice double helix for depicting DNA.

(C) Duplicate your zig-zagged line, and place it right beside the other line. Select your circle with the regular selection tool  , and fill it with a default gradient colour from the default colour palette. Open up the gradient window by going to **Window** > **Gradient**. Change the gradient from linear to radial, and reverse the sliders such that the window looks like this:



(D) Bring your lipid head and lipid tail together. Highlight the tails, right click, and select **Arrange** > **Send to Back** so that they render behind your lipid head. Select the whole lipid, and group it into one entity with **CTRL+G**. Begin to copy and paste them and arrange them slightly offset. After about 5 or 6 duplications, ensure the lipids on each end are aligned with each other.

(E) Group the whole motif and duplicate it. Rotate it 180 degrees, or reflect it over a horizontal axis using **Object** > **Transform**.

5 PyMOL Raytracing

Raytracing is the act of a computer trying to render an image that is reflective (ha, get it?) of real light shining off of an object. This is a computationally expensive process because it involves the casting of rays from the camera, or a light source, and tracking their path as they bounce off of things. As such, software will typically not raytrace in realtime, but rather give a computationally 'cheap' rendering. Once you have positioned and labelled your molecule appropriately, you can get a raytraced snapshot. It helps provide depth-of-field when looking at two-dimensional projections of three-dimensional objects, like protein structures. Let's start by opening up PyMOL and getting a protein structure loaded. In the console, issue the following command:

```
fetch 1blf
```

Get rid of the black background color by typing:

```
bg_color white
```

It doesn't really matter what your background color is when you raytrace, because we will set the command for a transparent background:

```
set ray_opaque_background, 0
```

However, having a white background gives you a better idea of how your color scheme will look on a white background; which is typically how things are printed or displayed in journals. We need to get rid of the awful default green colour. I prefer to set my entire molecule to white, and label the important parts with pastel colors. Selecting a color scheme is important for making your rendering look good.

```
color white, 1blf
```

I highly encourage making named objects while you work in PyMOL. It makes things so much easier. If there is an important motif in your protein, try this:

```
select lfcinB, resi 17-42
```

This creates a reference to this set of residues named **lfcinB**, which can be accessed from the user interface on the right hand side, or by its name via command line. Now try:

```
color slate, lfcinB
```

Any time you want to modify the colour of these residues, it is that simple! The C-terminal lobe of this protein consists of residues ∼341 to 689:

```
select c-lobe, resi 341-689
color salmon, c-lobe
```

Cool. Now, lets select the individual cysteine residues involved in the disulphide bond of the little lfcinB peptide:

```
select disulphide, resi 19+36
color wheat, disulphide
show sticks, disulphide
```

Note here that "disulphide" is the name I gave to the pair of residues 19 and 36, just as we have been doing. I didn't want it to get confusing as though I was 'selecting disulphides' by simply supplying the argument 'disulphide'. Now let's do the fun part - raytracing. I want the final figure to show the entire protein, as well as a close-up of the lactoferricin B peptide. The fastest way to ray-trace is to just type:

```
ray
```

This will take a few seconds to minutes depending on the scene complexity, and your computer's graphics card. Once it is finished, don't move the scene! If you do, you just have to raytrace again. Sure, the final product looks pretty, but it can look prettier. PyMOL has a few built in raytrace modes that give the molecule different looks.

set ray_trace_mode, 0	→	default
set ray_trace_mode, 1	→	normal + black outline
set ray_trace_mode, 2	→	black & white
set ray_trace_mode, 3	→	quantized color + black outline

I want to set quantized color, because I think Max Planck is cool.

```
set ray_trace_mode, 3
```

Now we'll position the molecule to the center of our screen, and type:

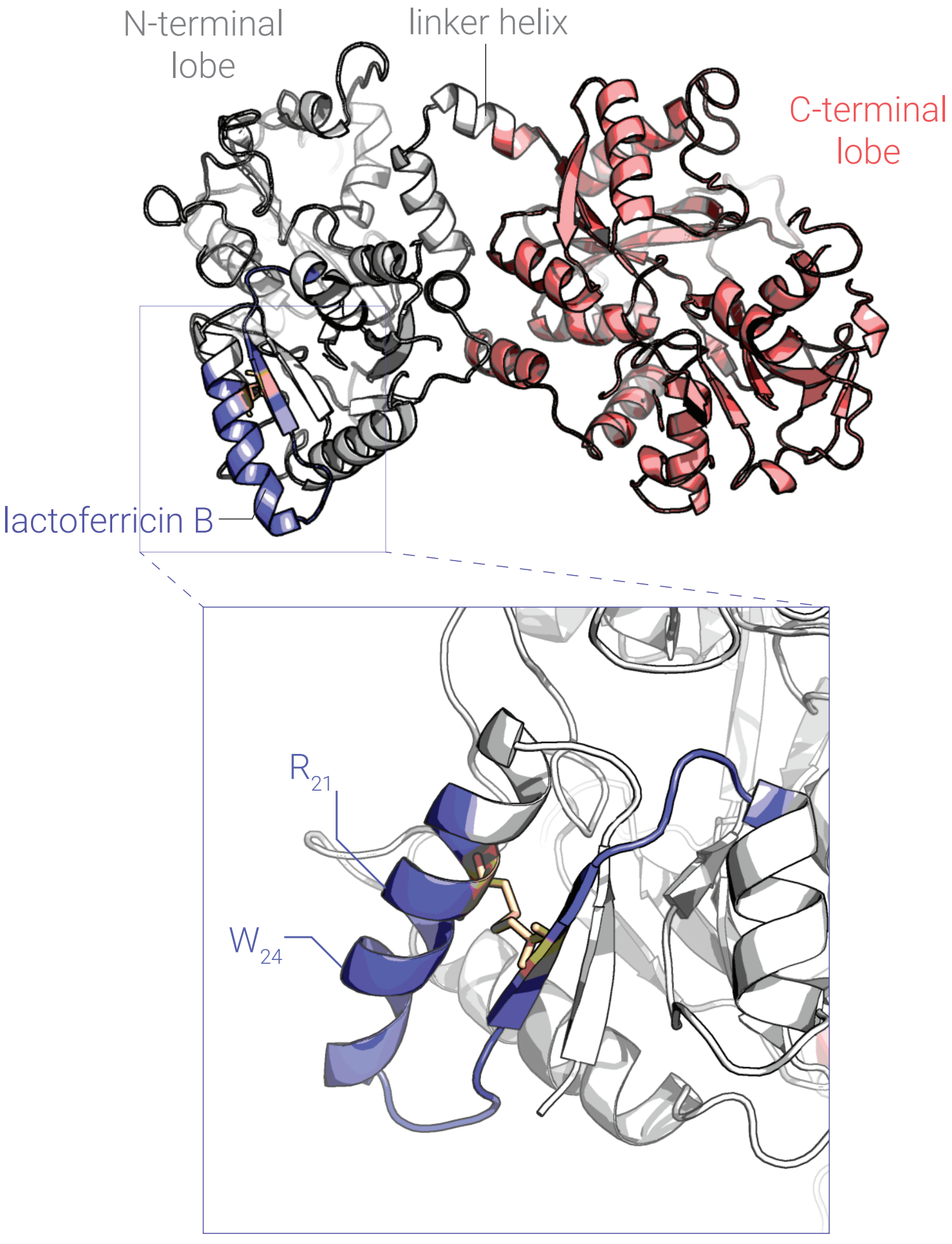
```
ray 800,800
```

The "800,800" sets the dimensions of the image. If you exclude this argument, it will default to the screenspace occupied by your viewport. You may find some part of the image is truncated if you do this, so you may need to reposition. The image can now be saved by going to File > Save as > PNG, and choosing the directory. If you're a nerd, and would rather save by command line, you can type:

```
png directory\filename.png
```

Now let's render one of a close-up of the blue peptide, and bring these images into Adobe Illustrator. Use the scroll-wheel on your mouse to alter the depth-of-field, and made the background less visible before raytracing.

Once both of these images are in Illustrator, you can play around with how you would like to present the data. I have given a sample final figure on the next page.


N-terminal lobe
linker helix
C-terminal lobe
lactoferricin B
R21
W24

6 Useful PyMOL Commands

Fetch a protein from the Protein Data Bank:

```
fetch <code>
```

Remove all the waters and heteroatoms from the coordinate file/scene:

```
delete hetatm
```

Show the crystallographic unit cell:

```
show cell
```

Expand the asymmetric unit to fill the unit cell

```
symexp myExpansion, <protein code>, <number of copies>
```

Select and name hydrophobic residues:

```
sele hydrophobes, resn PHE+TRP+TYR+ALA+VAL+LEU+ILE+MET+GLY
```

Set surface quality high:

```
set surface_quality, 3
```

Set opacity/transparency of an object between 0 and 1.0:

```
set transparency, 0.5, <object>
```

Create electrostatic surface map:

(A) Select the  action for your protein

(B) Click **Generate**

(C) Click **Vacuum Electrostatics**

(D) click **Protein Contact Potential (Local)**

Measure distance between two atoms:

(A) In the console window (not the PyMOL viewport), select **Wizard**.

(B) select **Measurement**

(C) Select the first atom.

(D) Select the second atom.

(E) You can continue selecting pairs of atoms in this way, and when you're done, select 'Done' at the bottom right of the PyMOL viewer window.

7 Blender Raytracing

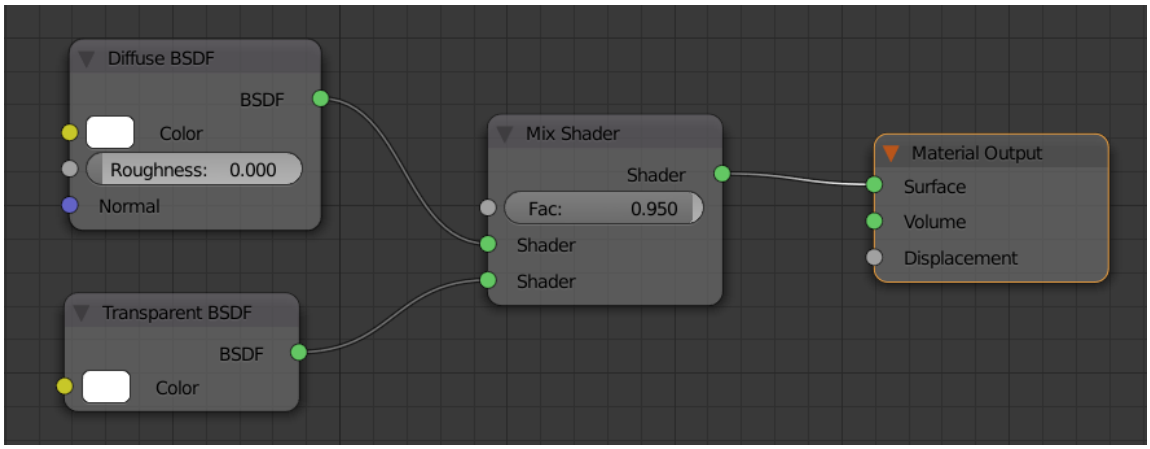
Ensure that you have VMD and Blender installed for this section. Blender is enormously complex, and has entire volumes of books devoted to learning its capabilites. I will not go in depth as to how to perform certain actions in Blender, rather I will assume you have played around and can carry out basic tasks. This section is more for advanced users of this software that want to get started with converting atomic coordinates to polygon meshes and applying materials before raytracing in professional animation software. Animations of course can also be done in Blender, but I will not cover this. Blender, nor any addons that I am aware of allow for PyMOL-style manipulation of atomic coordinates, but rather everything is based on vertex manipulation. Thus, it is important to be very precise when defining colours of different domains of your protein, or to reserve this task entirely to be done in PyMOL, as it is technically not scientifically accurate to 'paint' on your molecules. As such, I typically treat Blender renderings solely for illustrative purposes in presentations.

Addons for Blender are being developed that will allow for direct fetching of atomic coordinate files from the PDB, though they are still relatively unstable, so I prefer to do things manually. If you want to check out such addons, search Google for ePMV. With your PDB file of choice downloaded onto your computer, load up VMD and select File > New Molecule... and browse for the file of interest. Once you have selected it, press load to get it into the frame. Now go to Graphics > Representations... and in the Drawing Method dropdown select **New Cartoon**.

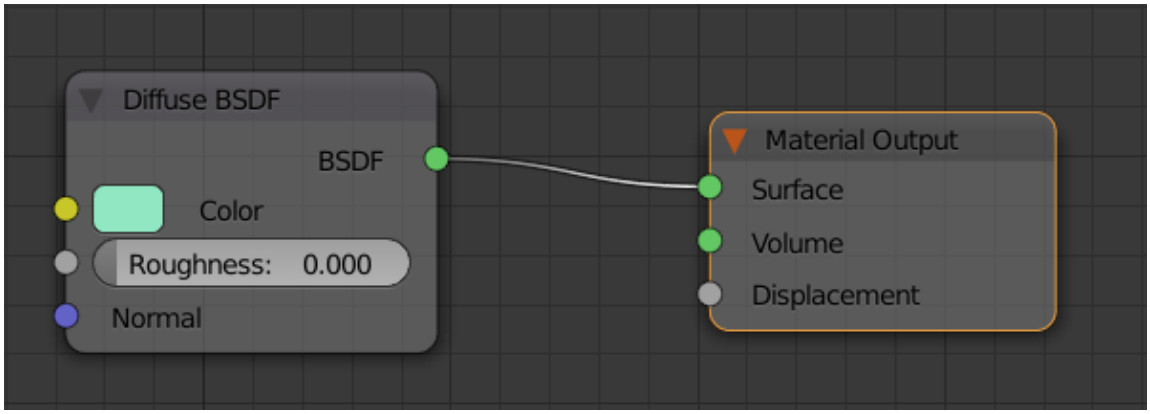
Ensure the viewport rotation is stationary by slowly rotating and letting go of the viewport with left mouse click. Render the protein to a polygon mesh by going to File > Render and selecting **Wavefront (OBJ)**. Without changing the viewport positioning, render another

file but this time with **Surf** selected as the Drawing Method.

These two files can now be imported into Blender. Change the rendering engine to be **Cycles Render**. Delete the existing materials on the objects. I usually change the existing lamp in the scene to a Hemi lamp, and increase its brightness accordingly. On the surface representation, I use the following material to stylize the surface:

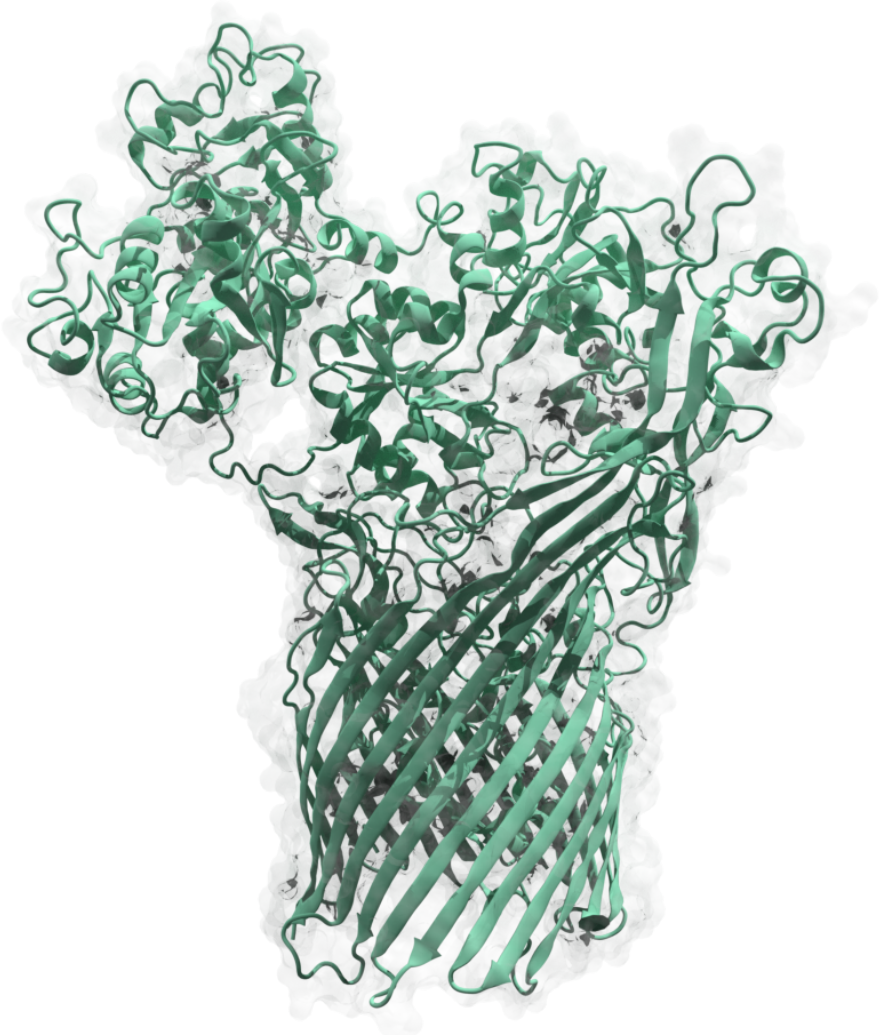


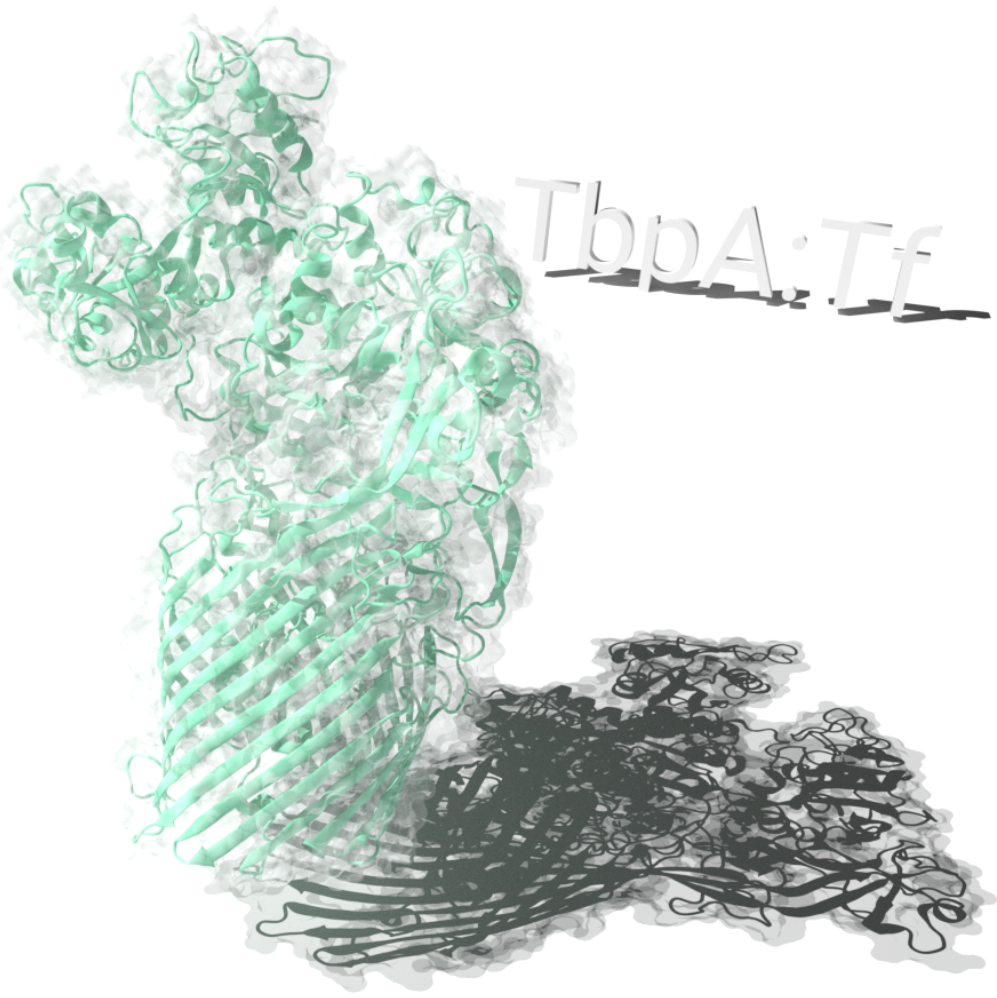
And for the cartoon:



Position the camera appropriately, and render a 2000px × 2000px image. To use your graphics card instead of your CPU, select Device: GPU Compute. Under Film, select Transparent. Under Performance,

specify 256px tile sizes for X and Y. Under Sampling, select 1000 samples for Render. Then press F12. Below are some samples of Blender rendering.



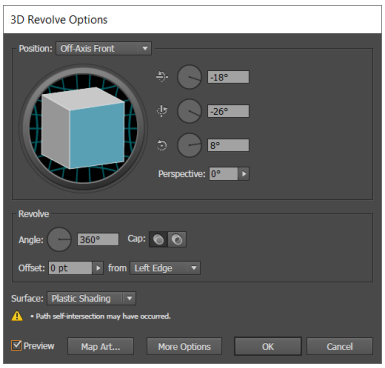

TbpA:Tf

8 Simple 3D Objects

Adobe Illustrator is very powerful for making cartoonized versions of things; even in three dimensions. You can draw almost any shape, and add a third dimension to it by selecting it, and going to *Effect* $\rightarrow$ *3D* $\rightarrow$ ...

1. Extrude & Bevel
2. Revolve
3. Rotate

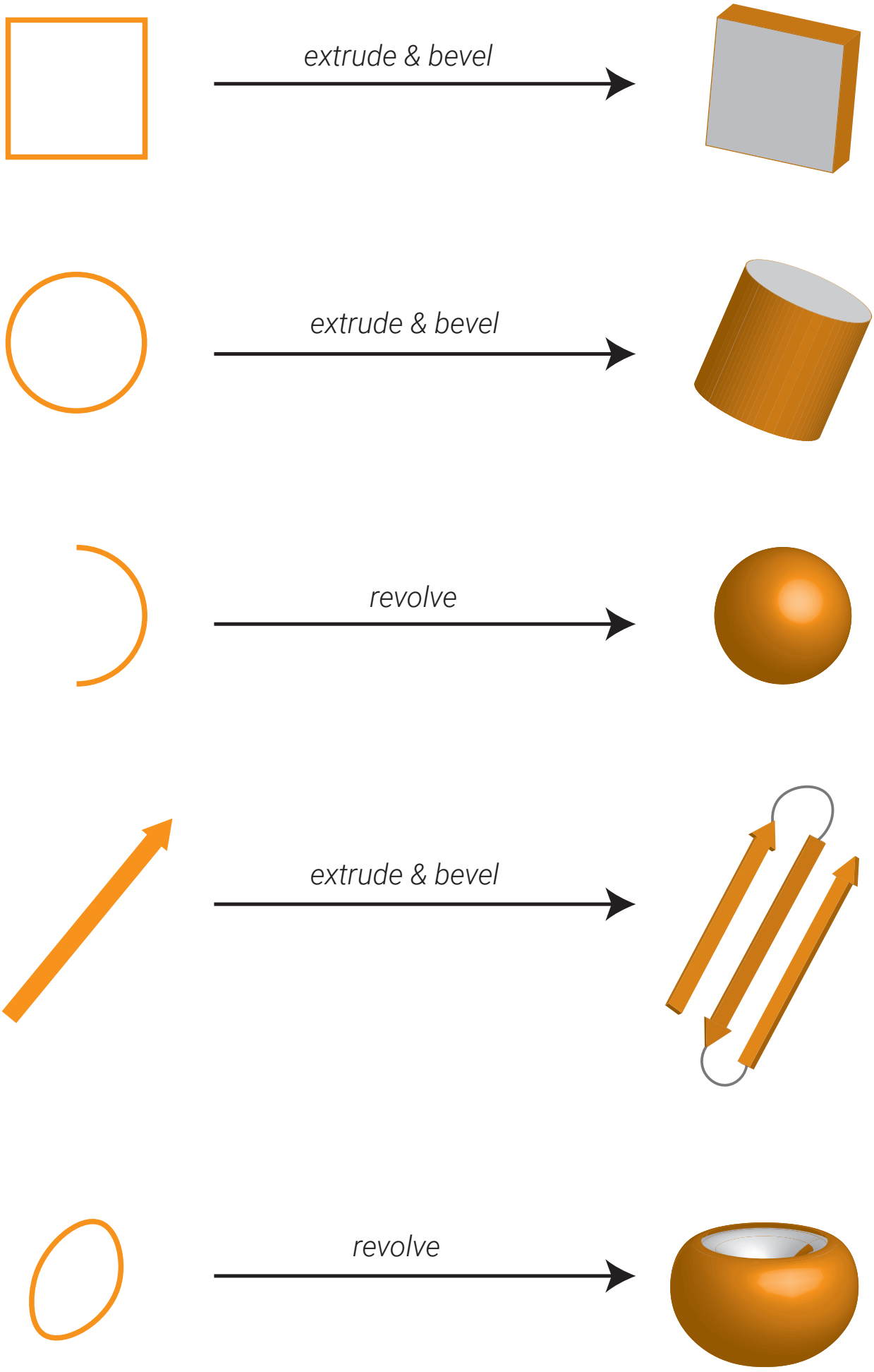
Extrude & Bevel will 'pull' your shape along an axis to create the third dimension. **Revolve** will spin your shape around a central axis. In this way you can create perfect spheres, and toruses. **Rotate** will create a projection of your shape from the perspective you desire. The following is the options window.



Firstly, make sure you select 'Preview' at the bottom left. This will render the object in real time. The 3D cube can be grabbed and rotated to position the object at the perspective you want. Angles can be adjusted manually in the boxes on the right. Offset will provide more or less depth to your extrusion. Feel free to play around with the

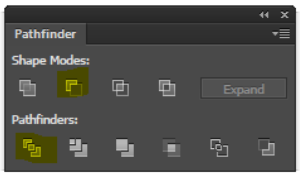
other options. I have given a few examples of things you can make on the next page.

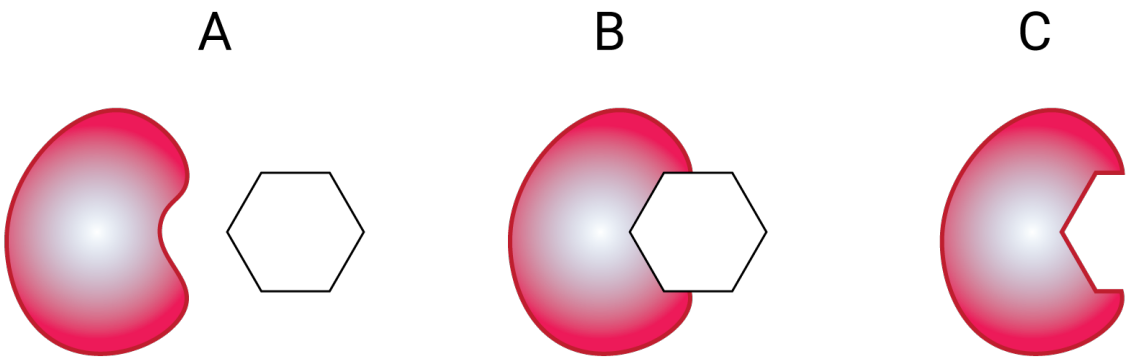
Manipulation of paths allows you to do a lot in Illustrator. If you want to make the sphere for example, you can make a simple circle, deselect it, select the direct selection tool  and then disconnect its top and bottom vertices by directly clicking on the anchor points and selecting  at the top of the window. This will allow you to separate the circle into two semi-circles. Delete the left hemisphere, and revolve the remaining object.


extrude & bevel
extrude & bevel
revolve
extrude & bevel
revolve

9 Lock & Key Modelling

Suppose you are trying to show a protein:ligand interaction. It is customary to draw a ligand binding site on the protein as a small shape removed from the entirety of the sketch. Adobe Illustrator has a **Pathfinder** tool window that is perfect for such applications. Ensure this window is open by clicking **Window** > **Pathfinder**:





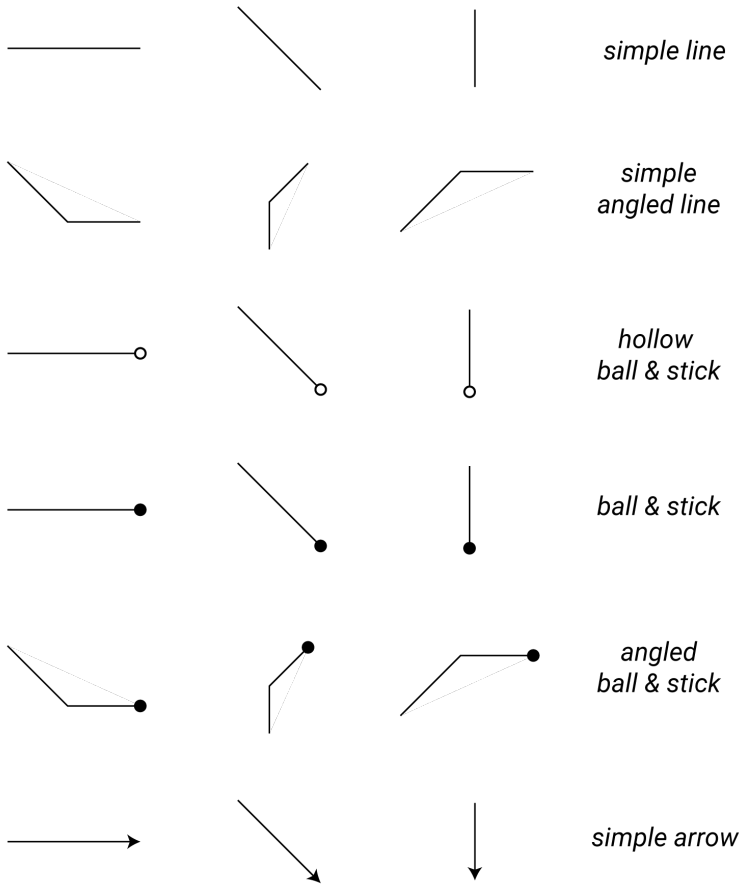
(A) Draw your protein (shown in red) and your ligand (shown as a plain white hexagon).

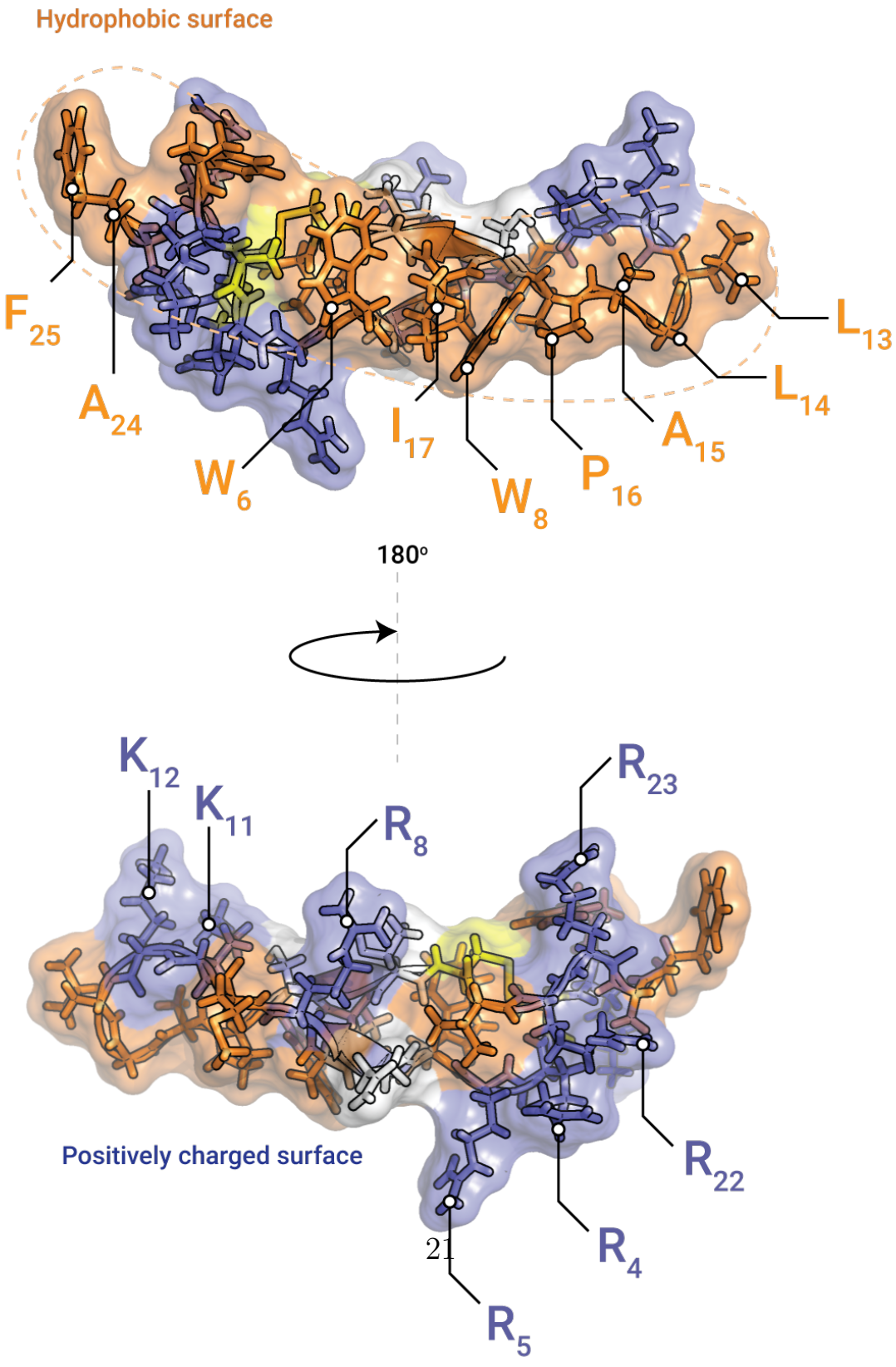
(B) Position them such that they are bound.

(C) In the pathfinder window, select **Minus Front** or **Divide**. If you choose Divide, you may need to ungroup the selection afterward with right click > Ungroup.

10 Label Lines

I really don't need to say much about label lines. They label things in your figures, and they are lines. I think an overarching theme in making scientific pieces visually appealing is consistency. As such, I keep all lines oriented at 90°or 45°. Below are some of the label lines I use in figures. The following page has an example of them in use.



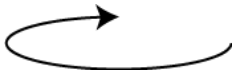

Hydrophobic surface
F25
A24
W6
I17
W8
P16
A15
L14
L13
180°
K12
K11
R8
R23
R22
R4
R5
Positively charged surface

11 Rotation Arrows

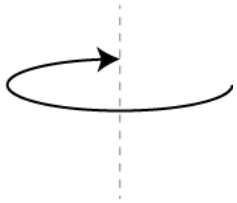
Let's take a look at how to make rotation arrows like these:

Start by using the circle tool to make a squished ellipse. Set the fill to null and the stroke to 1pt black  . Select the **Direct Selection** tool (which looks like a white cursor ), and deselect the circle by clicking somewhere in the whitespace of your document. Slowly move your mouse over the path of your ellipse to the top middle of it until it locks onto the anchor point. Select it, and then click **Cut Path at Selected Anchor Points**  . This will disconnect the anchor point from the path. Do this once more on the right-hand anchor point, so that you are disconnecting the 'upper right' segment of the ellipse. Now, you can select the disconnected path and delete it:

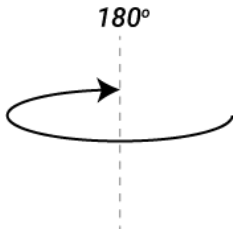
Now, select the regular selection tool  , and select the segmented ellipse. At the top of the window, click **Stroke** to open up the stroke options window. Here you can make arrowheads, and dashed lines. Make the right end of your segment any arrowhead of your liking:



At this point, draw a vertical line down the center of the ellipse. Give it an opacity of 50%, and make it dashed with a 6px dash width:



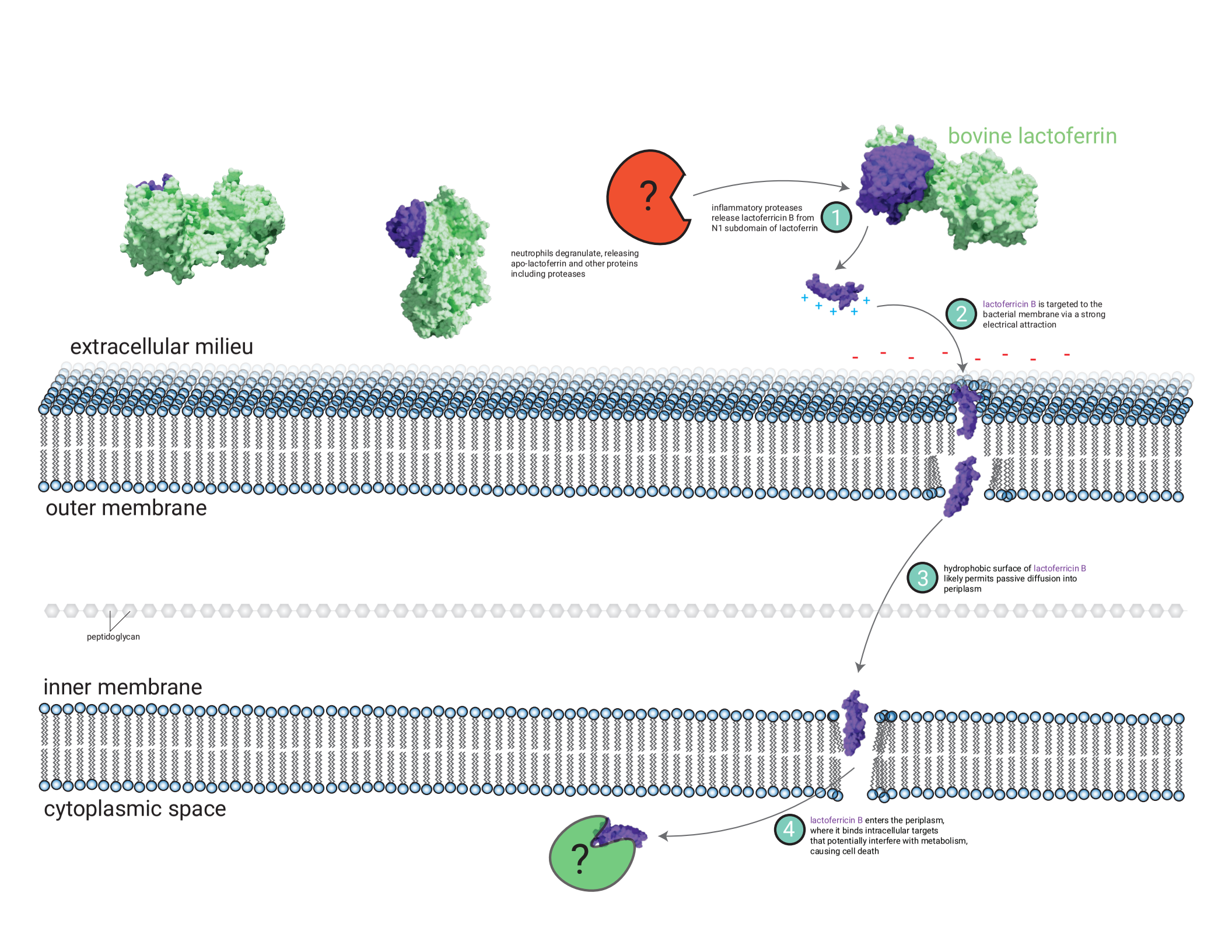
Finally, you can add the rotation degree to the top of your rotation axis. To make the degree symbol, just use a lowercase 'o', and then superscript it using the **Type** window:



12 Full Schemas

For a full schema, I prefer to use Blender-rendered objects, unless there is something very specific that requires labelling (i.e. specific residues). As a general note; multiple renderings of the same protein in different orientations adds a bit of realism to the schema.

On the following page I have given an example of a schema that utilizes some of the processes touched on in the previous sections.


bovine lactoferrin
?
inflammatory proteases release lactoferricin B from N1 subdomain of lactoferrin
1
neutrophils degranulate, releasing apo-lactoferrin and other proteins including proteases
2
lactoferricin B is targeted to the bacterial membrane via a strong electrical attraction
extracellular milieu
outer membrane
3
hydrophobic surface of lactoferricin B likely permits passive diffusion into periplasm
peptidoglycan
inner membrane
cytoplasmic space
?
4
lactoferricin B enters the periplasm, where it binds intracellular targets that potentially interfere with metabolism, causing cell death