EMAN2 Reconstruction Tutorial
Using the Workflow interface

Note that this tutorial was updated on 3/10/2011 to reflect any changes in the interface, etc. It should not be used with versions of EMAN2 prior to 2.01.

➡ EMAN2 documentation is largely provided via the Wiki at: http://blake.bcm.edu If you wish to edit the Wiki, first, create an account for yourself, then send email to sludtke@bcm.edu and we will adjust permissions so you can edit. Previously this was an 'open' wiki, but we had serious spam problems, and now have to approve new users. You don't need an account to view the contents, of course.

➡ **GUI Tips:** Interactions with the interface in EMAN2 are quite similar to EMAN1 in many ways. EMAN2 will work best with a 3-button scroll mouse, though there are alternatives using keyboard modifiers for people using one button mice on Macs.
  - In most display windows (plots, 2-D images and 3-D volume display), the middle mouse button will open a 'control panel' for the widget with many options to control the display
  - The right mouse button is used for panning in 2-D or 3-D image windows, and can be used to zoom (by shift+dragging), and to reset the zoom (clicking) in plot windows.
  - The left mouse button has various purposes in various contexts.
  - The scroll-wheel will generally act as a zoom. Use the control-panel for more precise control
  - If you have a one button mouse, one of the modifier keys (depending on platform) combined with a mouse click will serve the same role as a middle-click. You may need to try them (alt, command, ctrl, shift) to discover which works on your machine.
  - In the control panels, and other places in the EMAN2 interface you may encounter 'Value Sliders'. A slider is attached to a text-box with a number in it. Dragging the slider controls the number, and entering a number will change the slider. In addition, the text-box can be used to control the range of the slider and get more precise control. By typing '<value' or '>value' in the text box you can change the limits of the slider. Note that it is also possible to enter values which are outside the current slider range.

Introduction
EMAN2 can be used at many different levels ranging from high-level task-based workflow interface to writing code in C++ or Python. In this tutorial, we will be focusing primarily on the Workflow interface, which will guide you through the single particle reconstruction process, and keep track of appropriate intermediate data generated during the process in a self-consistent way.

We begin with a summary of the steps. We'll be bypassing steps 1 and 2.

1) Import and optionally preprocess micrographs/CCD frames
2) Picking particles
3) CTF determination & phase-flipping
4) Mark bad particles and build particle sets
5) Make 2-D reference free class-averages
6) Build an initial 3-D model
7) Coarse 3-D refinement
8) High resolution 3-D refinement
9) Map assessment

Getting started
  - This is a 'choose your own adventure' tutorial. We've provided 3 different data sets for you to choose from. To make the download tolerable, rather than providing the raw CCD frames/micrographs, you'll be starting with boxed-out particles. If you like, raw data for a range or projects is available from: http://ncmi.bcm.edu/publicdata/db/home, and you can box particles out to your heart's content.

The 3 data sets you can choose from are:

	Sufficient data for	Symmetry	Voltage & Cs	A/pixel	# Ptcl	Approx Mass
GroEL (film)	~8.5 Å resolution	D7	300 kV 4.1 mm	2.12	~5000	800 kDa
mm-cpn (CCD)	~8.5 Å resolution	D8	300 kV 4.1 mm	2.66	~5000	1000 kDa
e.coli Ribsome + SECY	~11 Å resolution	None (C1)	200 kV 2.1 mm	2.8	~10,000	2500 kDa

- Either of the chaperonins will give you sufficient resolution to see α-helices. The ribosome will achieve sufficient resolution to begin to resolve RNA helices. Due to the lack of symmetry and larger number of particles, the ribosome will take considerably more computation than either of the chaperonins.

- Once you have selected a data set to work on, from the command-line:
    - *cd* <groel/ribosome/mmcpn>
    - you should see a 'raw' directory inside this directory
    - *e2workflow.py*

➡ Windows users - While you could use the traditional icons to run EMAN2 GUI programs under windows, we strongly suggest using the windows command-line (as described for Mac/Linux). You will be able to more easily monitor error messages, and things are less likely to go wrong.

- If you did this step successfully you will have two windows : 'Running Tasks' and the other will be titled 'EMAN2 Tasks'. Running e2workflow.py in a directory will establish that directory as a Project Directory. All files associated with this project will then live in this directory or subdirectories of this directory. The system is designed so all of your data and results are self contained, so do not try to run e2workflow from another location, or try to keep your data spread all over the place. The goal is that at the end of the project you will be able to trace back any step you performed to produce any of your results without ambiguity. While you could use the normal command-line programs and produce reconstructions without this restriction, a year later, when you're publishing, and have to submit your model to the EMDB, you will appreciate the project system...

➡ EMAN2 is NOT conservative with its disk usage. Disk storage is so inexpensive now, we always tend to err on the side of keeping intermediate files. Make sure you have a lot of free disk space available when beginning any project. You can always delete things you don't need.

➡ PLEASE read this web page: http://blake.bcm.edu/emanwiki/EMAN2/DatabaseWarning

Setup Project and Import Data
The workflow interface is an expandable tree. Each level of the tree has a form with associated information or parameters, even the levels which just appear to be containers for the levels below them.
- Select **Single Particle Reconstruction** (not the + next to it)
    - This will open a window where you enter basic information about your project from the list above.
    - If you have 2+ cores, you can enter the appropriate number of CPUs. Ignore the memory field.
    - Click 'OK' once you have entered the parameters to save the values.
- Expand **Single Particle Reconstruction** by clicking on the '+' next to it
- Select **Particles -> Particle Import**, then press the **Browse to Add** button.
- Select the **raw** directory, then hilight all of the **.hdf** files and click **OK**
- The browser will close, and a list of all of the particle files should appear in the Import Particles window. Click **OK** in this window. Wait for importing to complete.

➡ When working with your own data, if you are, for example, importing boxed out particles from another software package, it is critical that you import the original, unmasked non-CTF corrected version of your particles. CTF correction in cryo-EM is not portable between software packages.

CTF

EMAN2 uses a substantially different CTF model than EMAN1. EMAN2 uses a data-based background curve. The edges of the images are used in determining the background noise spectrum. After determining the background automatic fitting of defocus and B-factor takes place. The uncertainty in B-factor tends to be fairly substantial, largely because it is only a very approximate representation of the true envelope function of the data coming out of modern FEG microscopes.

➡ For this process to work properly, and to have phase-flipping work optimally, it is important that the box size used for the particles is somewhat larger than the actual particles. This is the reason for using a box size 1.5 - 2x the largest size of your particle. It can be very painful to try to go back and change the box-size after going through the CTF step, so I urge you to use a good box-size from the start. This decision is already made for you in the tutorial, but keep this in mind when you work on your own data.

➡ When working with your own data, do NOT try to use CTF corrected particles from EMAN1 or CTF parameters from EMAN1. The new scheme in EMAN2 is vastly superior, and in most cases completely automatic. It will not take much time in most projects to refit the CTF.

CTF Fitting and Correction

• Start by running automated CTF determination on your particle data by choosing **CTF -> Automated Fitting**
• A form should appear with a table that lists the particles that you generated in the previous stage of the workflow. Other columns (which should be blank at this point) show any already determined CTF information.
• Check the **Auto high pass** checkbox. This will modify the SNR curve to eliminate the first sharp peak that appears in virtually all single particle data due to ice gradients and other issues.

➡ This sharp, low resolution peak can cause some issues leading to incorrect 2-D alignment, and hence classification. However, it is also responsible for some of the contrast which makes particles visible by eye. If working with your own data, note that it should NOT be used with negative stain or phase-plate data. For most other projects it is beneficial.

• Set **Oversampling** to 2. Make sure other parameters are correct.

➡ Amplitude contrast can only be determined experimentally through some rather tricky experiments, and frankly slightly different values will not have a strong impact in most reconstructions. However, for negative stain data, if you try CTF correction, you will likely want to use a much larger value than the default 10% (likely 60-70% will work better (?)).

• Select all of the images by either clicking in the upper left corner of the table, dragging with the mouse, or using shift-click. Then press **OK**. Numbers will start being printed on the console, and you should see processes appear in the 'Running Tasks' window. Fitting should only take a few seconds.
• Select the **CTF -> Interactive Tuning (e2ctf)** task. Make the resulting window a bit bigger so you can see all of the results. The table should now be fully populated. If any entries were not fit, go back to the previous step and rerun the fitting on them (this should not happen). You can sort the results by clicking on any column header. Survey the values quickly.
• Next, we want to double check the results manually, just to make sure it didn't get the defocus wrong on any of the images. For all 3 of the specimens provided in the workshop, the fitting should be near-perfect. You may find one or two images which need 'tweaking', and they should be fixed.
• Sort the data by defocus, select all of the images (again, clicking in the upper left corner of the table is the easiest way), then press **OK**.

- 3 windows will appear:
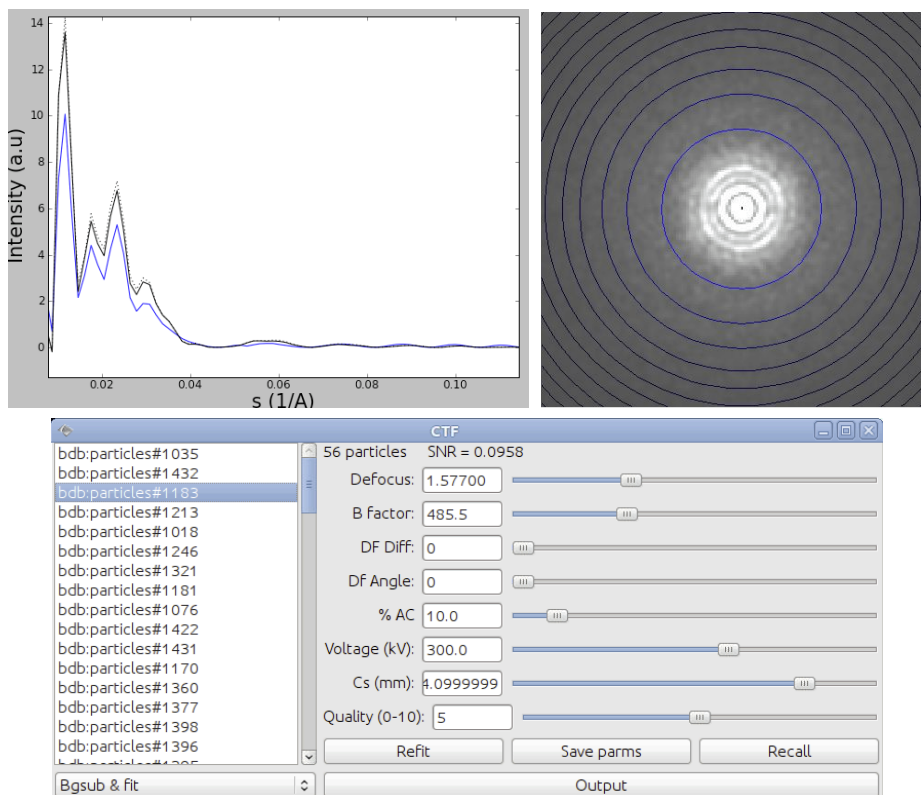
- Middle-click (or alt-click on Mac) will open a control panel for either of the top 2 windows, and permit you to make various adjustments. You can right-drag in the plot window to zoom in. Right click to zoom back out. If you change any of the CTF parameters for an image, you must press **Save Parms** to store the new values. The Quality slider is for your use only. The software does not directly act on this value. You can use this to mark images you think are better or worse by whatever criteria you wish. For this value only, you do NOT need to press **Save Parms** for the new value to be stores.
- The basic idea is for the blue (fit) line to match the black (data) line. The dotted black line is an alternate form of the black line which will permit better defocus estimation if a continuous carbon film was used on the grid, or if you have a particularly strongly scattering buffer. You will find that the curves do not match well at low resolution, due to not having a correct structure factor (yet). This is fine, at this point just try to make sure the defocus is roughly correct.
- I suggest just going through the images quickly by clicking on the first image in the list, making sure you can see the other two images, then using the down-arrow on your keyboard to go through each image in succession. Just make sure that the defocus is approximately correct for each image. If it is incorrect, use the slider to adjust it to something close to the correct value, and press 'Refit'. If this still doesn't give a good value, mark the image as lower quality and move on.
- The next thing we want to do is to create an estimated 1-D structure factor function from our data. *Do NOT select Generate Output at this point*. While you don't need to use all of the data for this purpose, as long as an image has the correct defocus, and there was nothing 'weird' about its power spectrum, you should include it. Close all 3 windows from the previous step and press **Generate Structure Factor**. Again, use an oversampling of 2. Select all of the images you want to use (4 absolute minimum), and press **OK**.
- Once the structure factor has been computed, we will use this to do a better fit of our data, so go back to the **Automated Fitting** step, select all of the data again, and press **OK**.
- Once refitting is complete, you should go through the fitting results one more time. You'll note that the black and blue curves will agree quite well now. You may still see some overall amplitude variations at low resolution, due to an attempt to get a better B-factor. It is more important that the amplitude of the fit curve be correct at high resolution than it be correct at low resolution. While you are welcome to manually adjust these parameters to obtain a better B-factor, etc, so long as the defocus is correct, the automatic parameters are fine for purposes of the tutorial.

➡ You might be tempted at this point to delete images which seem to be not quite as good as the others. The preferred approach in EMAN2 is to simply set them with a low quality. When you generate output from the CTF stage, you can then exclude any with low quality settings. This way, you have to option of going back later and experimenting with using some of this data, and also have some examples of 'bad' data around, which can sometimes be useful.

➡ Note that unlike in EMAN1, accurate per-micrograph B-factors are *not* critical for a good reconstruction or proper CTF correction (though it won't hurt).

➡ There are 2 cases where the automatic fitting is likely to fail A) very low resolution images where the first zero cannot be clearly seen due to a very large experimental B-factor and B) images where you have very few (<20-30) particles per frame.

• When you are happy with all of the fits, select the **Generate Output** task. In the subsequently appearing form select all of the images. If there are some with a lower quality, you can exclude them at this point (ctrl-click or cmd-click) if you like. Oversampling -> 1. Check all 3 boxes. Then press **OK**.

➡ Regardless of what oversampling value you used earlier, use an oversampling of 1 in the generate output step. If you do oversampling in this stage, it will make the phase-flipping process irreversible, and reduce some options you may have at later points in the processing.

Getting rid of bad particles and selecting which images to use

• When the generate output has completed, select **Build Particle Sets**. Then select **Wiener Filtered** in the panel that appears, and press **OK**.This interface serves a dual purpose. First, it allows you to manually identify 'bad' particles, and second, it is used to generate 'sets' of images which are then used for reconstruction.

➡ Selecting Wiener Filtered here does NOT mean that you will use Wiener filtered particles for your reconstruction, but simply that when you are manually picking bad particles on the screen, the particles it will show you are Wiener filtered.

➡ Particle sets allow you to try doing reconstructions with various subsets of your data without taking large amounts of disk space, and keeping track of exactly what you're doing. For example you may ask 'do I get a better reconstruction if I use only the 10 best images, or if I use all 20 images'.

➡ In general, one 'bad' particle can do far more harm than a single 'good' particle would do to aid the reconstruction. For purposes of the tutorial, the GroEL data set in particular has a number of particles which were boxed off the edge of the micrograph. There are a few bad particles in the other sets as well. While not absolutely necessary for the rough structures we are doing at the workshop, in general, you should get rid of bad particles either here or when picking.

• You will now see a window showing statistics for all of the images you have processed so far. Double-click on one of these images, and a tiled image display window will open. This window will show Wiener filtered versions of your particles. Left click on any particle to mark it as bad. Left click again to change your mind. A blue square will indicate that a particle is bad. A count of bad particles will also appear in the image list.
• When you have finished marking bad particles on the first image, you do not need to close the tiled image display, just double-click on the next image in the list, and the same window will be used.
• Once you have marked as many bad particles as you like in all of the images, you are ready to make your first set. Close the window displaying particle images, but leave the 'Build Particle Sets' dialog open. In the list, you should now highlight all images you want to use for your first reconstruction. Give the set a name like 'set-all', and click **OK**.
• When this is done, we also want to make a second set with a smaller number of particles to make the initial model building process faster. Open the **Build Particle Sets** dialog again. This time select only a few of the best images. You should target 1000-1500 particles in the set. Fewer will be required for GroEL and mm-cpn to

get good results, due to symmetry. More will be required for ribosome. Give this set a different name, like 'set-small', and press **OK** again.

➡   There are several different schools of thought over what will produce the best reconstructions, but there are a few general guidelines. First, if you are targeting high resolutions (towards 4-6 A or beyond), images too far from focus must be removed because the CTF phases cannot be corrected accurately past a certain limiting resolution. Second, while including images with very low SNR values will undoubtedly slightly improve the measured 'resolution' of your reconstruction, if they are too noisy they will actually just be contributing noise/model bias to your structure rather than actual useful information. As a general rule of thumb, if you can't see the particles fairly clearly in the Wiener filtered data, it's probably too noisy to use. Beyond this, you may need to experiment to determine the optimal data subset to use for your reconstruction.

Generating reference free class averages (2D refinement)

• Under **Reference Free Class Averages**, select the single entry : **Generate Classes – e2refined2d**

➡   Note that if you click the **Reference Free Class Averages** task itself you will see a form that lists the reference free class averages currently associated with the project. You can add your own reference free class averages to this form using the usual mechanisms (Browse To Add etc). This can be useful if you want to try generating an initial model using class averages generated outside the workflow (with XMIPP, Spider, etc.).

• You will see a small form asking you to choose from your regular, phase flipped and Wiener filtered particles. Choose the phase flipped-hp data and hit ok. This will pop up a form with three tabs asking you to specify the parameters required to run e2refine2d.py. The default parameters will produce suboptimal, but usable results, and it will run quickly. However, we need to make a few data-set specific changes:
   • GroEL: **# classes** =12, **alignment references** = 3
   • mm-cpn: **# classes** = 24, **alignment references** = 4
   • Ribosome: **# classes** = 24, **alignment references** = 8
• Then select the '**small**' set from the list, and if you have multiple cores enter 'thread:2' under **Parallel**(or however many you have) and click **OK**.

➡   The shrink option on the Simmx page: For the reference free class averages you can shrink the data to save time. The time savings can be quite substantial with larger shrink values, but the quality degrades as well. In some cases, the next step (initial model generation) will work quite poorly with images shrunken by too much. We are using the default value of 2.

➡   There is documentation text at the top of each panel in the dialog. If you want to obtain more information you can also go to the command line and type e2refine2d.py -h, or look at the wiki.

➡   If you want more information on the particular parameters you can pass to the aligners and/or comparators (in the Simmx and Class Average tabs) go to the command line and type 'e2help.py aligners -v 2' or 'e2help.py cmps -v 2'. Omitting the '-v 2' will produce a less detailed listing.

• Once e2refine2d has finished you will want to view the reference free class averages that were generated. Choose the Browse option in the EMAN2 tasks window. When the browser pops up you should see a folder called r2d_01. Note that if you have run e2refine2d several times you will most likely see several folders that start with 'r2d_' . Navigate into the most recent r2d entry (mostly likely r2d_01 at this stage). The reference free class averages are called classes_init, classes_01, and classes_02 etc. The highest numbered classes file contains the final results. Alternatively, you can look at the allrefs_XX files, which contain the same images, but sorted and aligned. Sometimes some of these images will be misaligned, though.

➡   For most projects, this is an ideal point at which to look for structural heterogeneity in your data (not an issue for the workshop sample data). If you see several class averages apparently in near-identical orientations, but with subtly different internal features, this may be a sign that your particle is moving in solution. For

example, a molecule like mammalian fatty acid synthase can be observed to move as much as 50 A in solution. It can be difficult in some cases to distinguish between variability and differences in orientation, so you may not get a definitive answer to this question at this stage. If you do observe what appears to be structural variability, keep it in mind as you move on to subsequent steps.

- Our goal now is to select class-averages to use for building an initial model. We don't want to include any obviously bad averages, and want to make sure we have representatives of as many different orientations as possible. Middle-click on the the tiled image display and select 'Del' in the control panel. Now click on any averages you think are bad. They should get covered up by a blue square. When you have deleted all of the particles you dislike, select **Save** in the control panel, and write the images to "**good-classes.hdf**". You can close the classes display then.
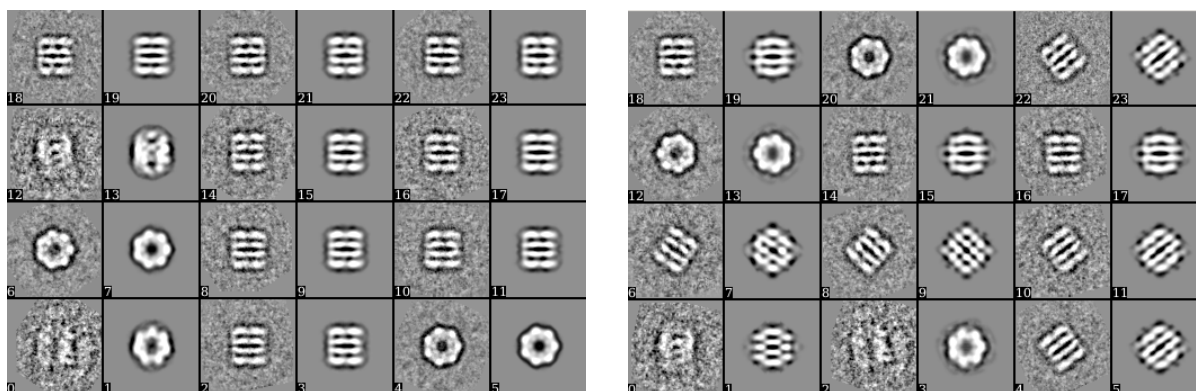
Making an initial model

There is a lot of controversy in the cryoEM community on this point. Some people feel that initial model generation is the most critical step in refinement, and you need to use difficult and time-consuming experimental methods such as random conical tilt or +-45 degree tilt methods to get an initial model before you can proceed. In the vast majority of cases, we have shown that the simple approach used in EMAN2 can give a completely reliable initial model with no additional experiments required. However, there are a few caveats here:

- ➡ Handedness cannot be determined from single particle data without tilting. If determining handedness directly from your data is important, and you aren't at high enough resolution for local folds to help, you will have to do some sort of tilt experiment. Personally I find single particle tomography to be the most appealing approach, since it is becoming very standard in many labs nowadays, and issues with direction of rotation, etc. likely have already been dealt with. You simply need to collect a tomogram with sufficient particles that the handedness can be observed. We will learn the basics of single particle tomography in a later tutorial. Random conical tilt is yet another choice, and may be necessary in a few situations, EMAN2 includes an experimental program for doing this as well.
- ➡ Tilt data can be used for validation of your results. Richard Henderson will speak to this point later in the workshop. While the tool for this is not yet a part of EMAN2, we do plan to incorporate this concept in the future.
- ➡ Heterogeneity is an issue. If you have a particle that is highly heterogeneous, the EMAN initial model strategy is likely to fail to produce a unique answer (since there isn't one). Single particle tomography may offer the best solution towards understanding the heterogeneity in your specimen. Once you understand the heterogeneity, EMAN2 includes e2refinmulti.py and e2classifyligand.py for purposes of processing such data sets using normal single particle data.
- ➡ Poor angular distribution. If your particles have a strongly preferred orientation, especially if this is combined with a low symmetry, there may not be enough information to produce an unambiguous starting model. However, it is also important to note that in this situation, even if you get a good starting model, refinement will also tend to degrade rather than improving the model. To perform a proper 3-D reconstruction, you must have a reasonable number of particles in orientations covering a *great circle* around the unit sphere.

- Of the 3 data sets provided for the workshop, ribosome is the easiest to determine an accurate initial model for. GroEL is next, and mm-cpn is the hardest. For GroEL the main issue is preferred orientation combined with the near square shape of the structure. For mm-cpn, the problem is that the structure is only slightly elliptical, and only has prominent features near the equator. However, as you will see, identifying bad starting models can be quite straightforward, and in most cases the correct starting model will be obvious, even without prior knowledge of the shape of your particle. In the case of GroEL or GroEL-like molecules, shrinking the data during 2-D class-averaging is particularly detrimental, as it can make incorrect starting models appear to be more reasonable. The 2-fold shrinking we have done in this tutorial was done for speed only. It will make getting a correct initial model more difficult.

- Under the **Initial Model** entry in the workflow, you should see **Make Model (e2initialmodel)**.

- ➡ If you wish to use an externally generated initial model instead of this process, you can add it/them using the **Initial Model** table using the 'Browse To Add' button.

- Choose **Make model (e2initialmodel)** and hit OK. This will pop up a form displaying the available reference free class averages. Click on **Browse to Add**, and select the **good-classes.hdf** file you created in the previous step.
- Enter the symmetry for your particle from the table at the beginning of the tutorial. 6 iterations is probably sufficient for most of the sample data sets, and you can start with 5 tries. If you don't get a good model on the first pass, you can run this step again to produce another 5 models to check. Click **OK** to start the model builder.
- Once the process is complete, open the Browser again, and go into the **initial_models/bdb directory**. You will see a number of model_XX_YY_zzzz files. The model_XX_YY files contain the 3D structures themselves. Double click on model_01_01 to view it. This should be your best model. Middle-click as usual to bring up a control panel and vary the isosurface threshold.
- To check the probable accuracy of our structure, double-click on the **model_01_01_aptcl** file. This file contains pairs of images: class-average, reprojection. Ideally these pairs should all look identical, and the 3-D structure should look reasonable. Here is an example of a good (left) and bad (right) aptcl file for GroEL:

- Having a good looking aptcl file is not absolute proof of having a correct initial model, particularly in the case of strong preferred orientation, but incorrect structures do tend to be quite obvious, particularly in 3-D. Hopefully your _01_01 model looks at least something like your structure should look. If not, it is possible (though unlikely) that one of the higher numbered models will be better. Otherwise if you don't have a model you trust yet, you should run the initial model generator again until you do. Note that the results of the second run will be _02_01, etc.

➡ GroEL is an excellent demonstration case, as it has so many possible failure modes, particularly if a large shrink value is used (due to it's near square shape and strongly preferred orientation). If you don't shrink the data at all, these failure modes will be much harder to observe (a good thing normally). Interestingly, an asymmetric particle, like ribosome, will work virtually 100% of the time using this method, even with a lot of shrinking) because of its distinctive shape. Regardless, as you can see, detecting 'bad' answers is generally pretty trivial.

3D Refinement

- We are now ready to do our first (coarse) 3-D refinement. This is the place where we have a lot of options to set and understand (particularly when we're going for higher resolution). Select **Run - e2refine** from the workflow. The first window will show a dialog for selecting Particle Data and Usefilt data. Leave the **Usefilt** data option set to **None**, as this feature has been unreliable in recent releases. Set the **Particle Data** to **Phase flipped-hp**. This says that you will be using phase-flipped high-pass filtered data for your reconstruction. Press **OK**.
- Now you will see an intimidating looking panel with many tabs at the top. We will need to set options in each of these 6 tabs before our refinement can proceed.

➡ Note that there is a reasonable amount of documentation at the top of each of the tab selected panels. While this won't explain all options, it will explain a fair amount about what you should select and when. The Wiki is generally the best place to look if you need more details.

- Particles tab:
  - Select which set of particles to use for your refinement. I suggest using the '**small**' set at this point. We will move on to the 'all' set for high resolution refinement.
  - Set **refinement iterations** to 5.
  - **Parallel** should have the usual **thread:2** entry.
  - Check **Low mem**
- Make sure the other values are correct, then move on to the Model tab:
  - Select the initial model you thought was best
  - **Auto mask 3D** should be checked, with the default parameters set
- Next, the Project3D tab:
  - Select your symmetry and value (eg - d8, d7, c1)
  - Enter a value: (mmcpn = 2.81, GroEL = 2.571, ribosome = 6)
- Simmx tab:
  - shrink =2
  - 2 stage simmx = 2
  - Ribosome & mm-cpn:
    - main comp = frc options -> 'snrweight=1:zeromask=1'
    - align comp, and refine align comp = ccc
    - refine aligner = refine
  - GroEL:
    - others same as ribosome
- Class Averaging tab:
  - averaging iterations = 2
  - class separation = 5
  - keep = 1.5
  - normalize.edgemean
  - averager=ctf.auto
  - check 'set proj sf on avg'
  - main comp = frc options -> 'snrweight=1:zeromask=1'
  - align comp, and refine align comp = ccc
  - refine aligner = refine
- Make3d tab:
  - reconstructor = Fourier
  - iterations = 2
  - pad to = 192
  - keep = 1.5
  - set sf checked
  - post processor = filter.lowpass.gauss   options -> 'cutoff_freq=.07'
- Double check all your settings, since this will run for some time and you won't have a second chance. When you're ready, hit **OK**.

This will run for some time. Likely we will be letting this run overnight, and checking our preliminary models tomorrow.