**Run with Symmetry:**

-the protocol has to adopt symmetry, here are the needed steps:

1. Generate a symm def file
2. ^if we need modify the symm def file, usually involves changing the set\_dof lines
3. Add symmetry related flags
4. Make sure to use the binary silent files format

**Breakdown of the steps:**

1. **Generate symm def file:**

The main input is a single monomeric structure, creating the symm def file of this single monomeric structure depends on what kind of file we start working with:

1. Using a symmetric (or near symmetric PDB file) PDB file:

Run the application: **make\_symmdef\_file.pl** has been included in src/apps/public/symmetry

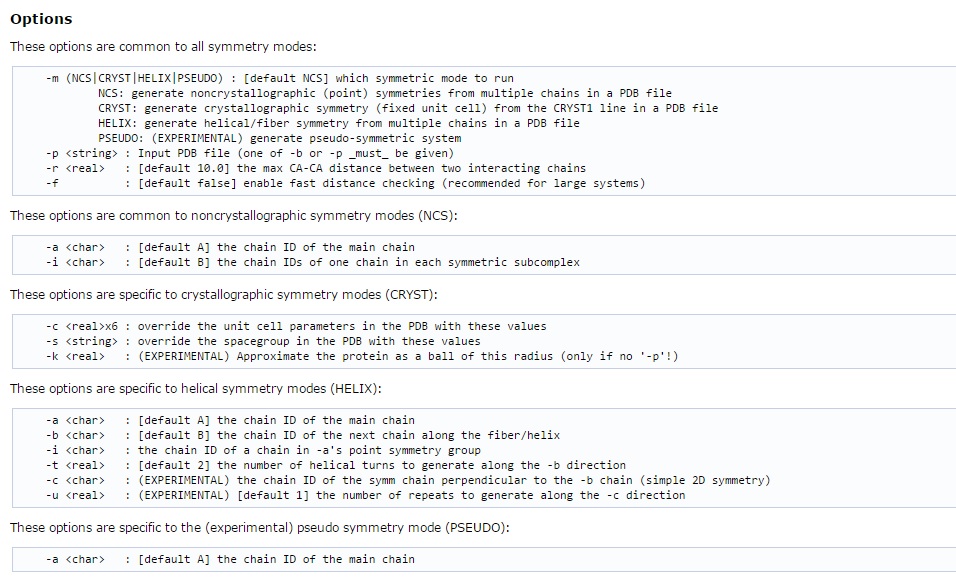
**Make\_Symmdef\_File:**

Main Goal of Application: "symmetrize" our inputted PDB file

3 different symmetry types: noncrystallographic (point) symmetries,

crystallographic symmetry, and helical symmetry.

Nonsymmetry type: pseudo-symmetry,



Link: <https://www.rosettacommons.org/manuals/archive/rosetta3.4_user_guide/db/d1b/make_symmdef_file.html>

OR

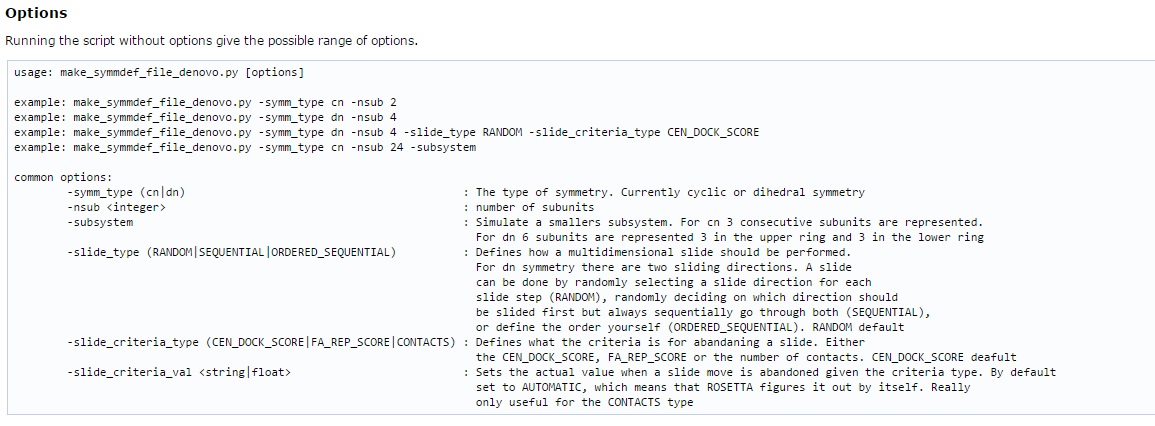
1. Starting from scratch, typically for denovo prediction (<structure is predicted from its amino acid primary sequence):

Run the application: **make\_symmdef\_file\_denovo.py** -- has been included in src/apps/public/symmetry

**Make\_Symmdef\_File\_Denovo:**

Main Goal of Application: "symmetrize" our inputted file based on a specified symmetry

Specified symmetries have to be made by hand:



Link: <https://www.rosettacommons.org/manuals/archive/rosetta3.4_user_guide/d4/d4a/make_symmdef_file_denovo.html>

1. **Modifying (only if we need to! check before hand):**

Relax: “Relax” is an application in Rosetta that carries out the task of remodeling full atom Rosetta models into simple structures (Also reads Centroid models)

- preparing our structures depends on what we are going to use them for

- for this reason “relax with all-heavy-atoms constraints” aka “relax” was created

Link about Relax:

<https://www.rosettacommons.org/manuals/archive/rosetta3.4_user_guide/d6/d41/relax_commands.html>

1. **Add symmetry related flags: (this is the part involved in symmetric docking)**

code for the symmetric docking application is in: rosetta/rosetta\_source/src/apps/pilot/andre/SymDock.cc

example of symmetric docking protocol and input files: rosetta/rosetta\_tests/integration/tests/symmetric\_docking

Flags are like trackers or side notes they inform you of basic things that you would probably want to know, so a standard run would involve the following flags

-in:file:s

-in:file:native

-database

-symmetry:symmetry\_definition

-packing:ex1

-packing:ex2aro

-out:nstruct

-out:file:fullatom

-symmetry:initialize\_rigid\_body\_dofs

-symmetry:symmetric\_rmsd

Link on docking symmetric: <https://www.rosettacommons.org/manuals/archive/rosetta3.4_user_guide/d4/dae/symmetric_docking.html>

1. **Binary Silent Files Format**