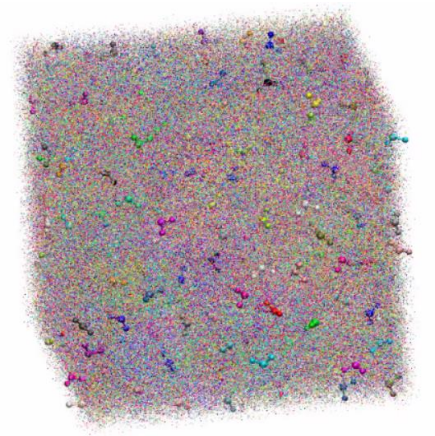
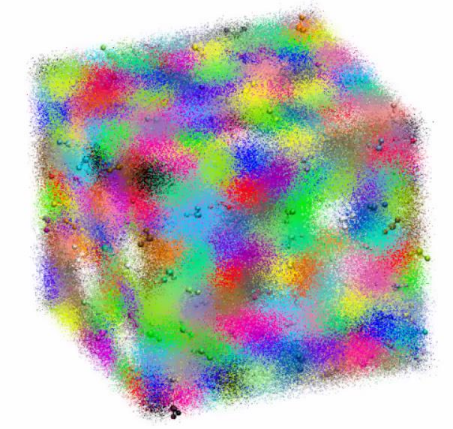
Installing g_permute plugin for/and GROMACS

g_permute is a tool to relabel the solvent molecules in a molecular dynamics trajectory such that their distance to a reference position becomes minimal. The result is a new trajectory, where the solvent molecules are centred around their reference positions, rather than exploiting the full configuration space as in the original trajectory. This procedure is called permutation reduction.



Unpermuted trajectory



Permuted trajectory

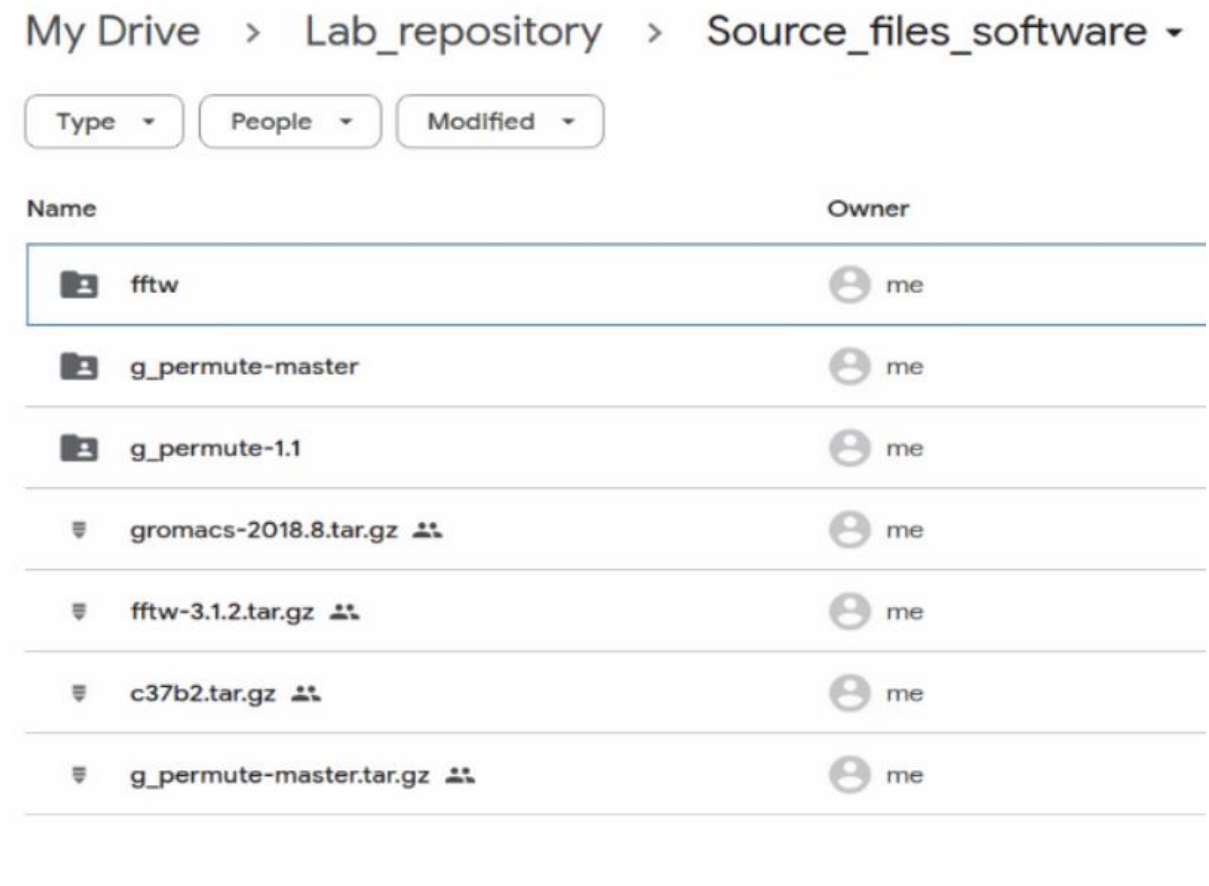
Installation procedure:

- g_permute tar file can be obtained from the GITHUB link https://gitlab.gwdg.de/lheinz/g_permute
- g_permute, available in the latest version works only in GROMACS 2018 version and is compatible with ubuntu version 18.04.
- Higher version of Ubuntu and miscompatibility of GROMACS version may not allow g_permute installation properly.

1. Install GROMACS with the following command lines.

tar -xvf gromacs-2018.8.tar.gz
cd gromacs-2018.8
mkdir build
cd build
cmake .. -DGMX_BUILD_OWN_FFTW=ON -DCMAKE_INSTALL_PREFIX=/home/hamsa/linux-apps/GMX/2018.8 -DBUILD_SHARED_LIBS=ON
make
make check
sudo make install
source /usr/local/gromacs/bin/GMXRC
Make sure the shared library is on, else g_permute will not work.

2. Before installation of g_permute, make sure fftw libraries are generated and located properly. You can install fftw from tar file or copy the already created fftw folder in the /home/hamsa/linux-apps/ path (folder is available in the following path)



If you want to create the fftw folder using tar file, follow the command.
tar -xvf fftw-3.1.2.tar.gz (You may also try fftw-3.3.1.tar.gz)
cd fftw-3.1.2
./configure
make
make install

3. Now install g_permute using the commands
tar -xvf g_permute.tar.gz
cd g_permute
cd src
More detailed debugging procedure is given in the following file.
/home/hamsa/linux-apps/g_permute/INSTALL

Edit the MAKE file with the appropriate path for the following

```
#Makefile for g_permute; edit the variables according to your needs

# load the following modules when compiling on owl
#module load shared
#module load binutils/2.25
#module load gcc/6.4.0

SHELL=/bin/sh

# set LD_LIBRARY_PATH
```

```
#export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:

# this is where executable and libraries will go - we recommend to
# leave PREFIX unchanged
PREFIX  = ../../g_permute-1.1
EXEC_PREFIX = $(PREFIX)
BINDIR  = $(EXEC_PREFIX)/bin

# pointer to your Gromacs installation
GMXDIR  = /home/hamsa/linux-apps/2018.8
GMXLIB  = $(GMXDIR)/lib
GMXINC  = $(GMXDIR)/include

# we have to link fftw, even if we don't use it
FFTWLIB = /home/hamsa/linux-apps/fftw/lib
```

Now give the commands
make
make install

**********g_permute is installed now**********

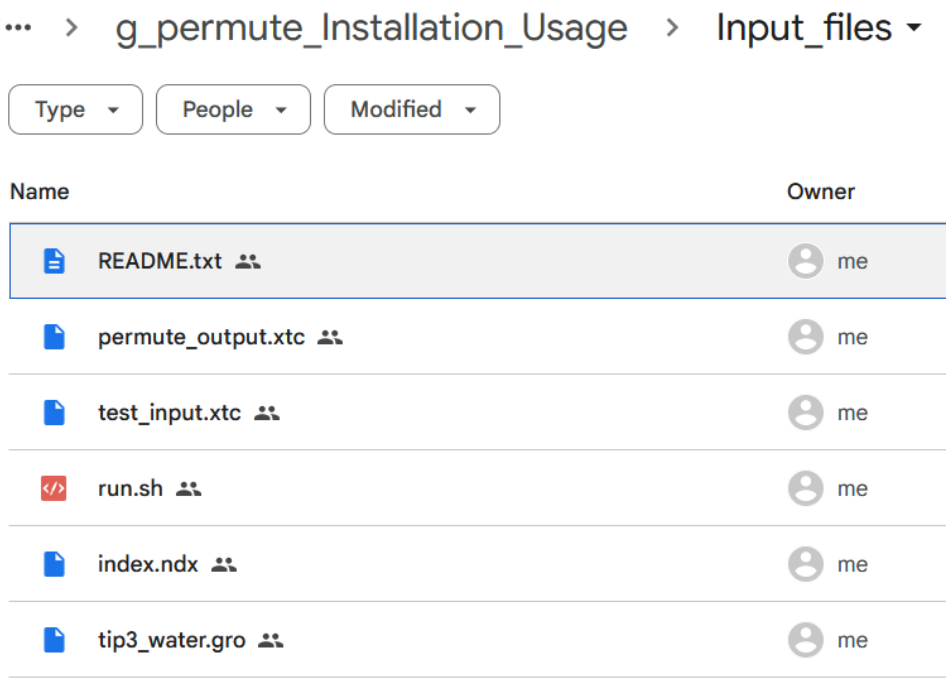
4. Copy the gmx executable to /usr/local/bin using the command
cp gmx/usr/local/bin

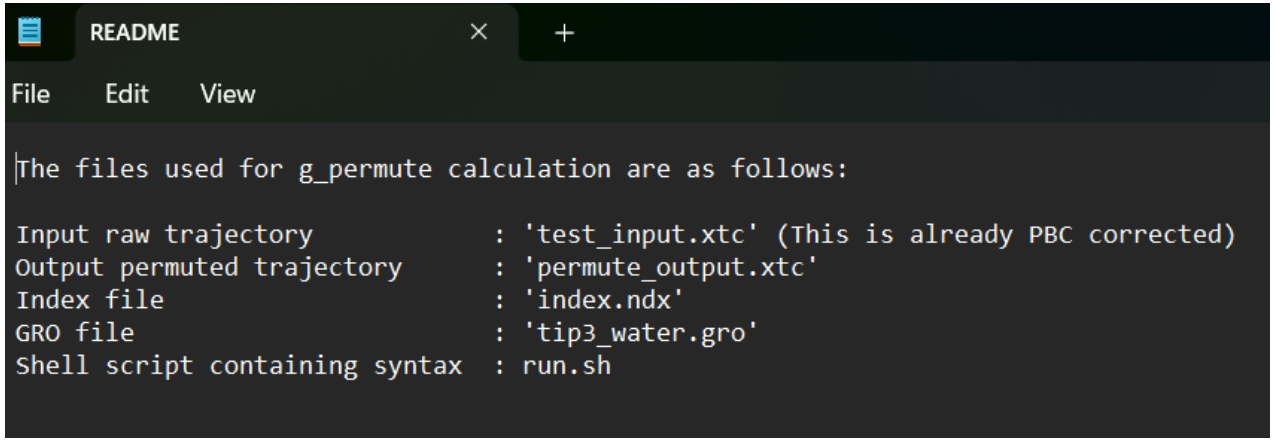
Points to remember during installation – Troubleshooting procedures:

1. *g_permute* was obtained from the GITHUB link https://gitlab.gwdg.de/lheinz/g_permute
2. This version is compatible only with *gmx 2018.8* and no other source files for g_permute is available currently. Hence g_permute will not work in lower version of GROMACS.
3. Higher version of GROMACS (beyond 2020) has compatibility issues with the ubuntu OS and may not work efficiently, hence GROMACS 2018 is recommended for g_permute usage.
4. Even if GROMACS2018 is installed in your system, make sure you re-install it properly using the above given commands.
5. Also, make sure the shared libraries are enabled while installation of GROMACS, else g_permute could not be properly installed.
6. The path to GMX libraries should be properly sourced.
7. The procedure is given in the link https://gitlab.gwdg.de/lheinz/g_permute/-/blob/master/INSTALL

g_permute usage

Before referring the subsequent slides, please take a look at the README document.





g_permute usage

Once g_permute is installed, perform the following steps to permute the trajectory.

Step1: Remove the PBC of the raw trajectory

gmx trjconv –f md.xtc –s md.tpr –o test_input.xtc –pbc mol –ur compact

Step2: Create index file with two groups: water oxygen and all water molecules

gmx make_ndx –f tip3_water.gro –o index.ndx

```
Command line:
  gmx make_ndx -f tip3_water.gro


Reading structure file
Going to read 0 old index file(s)
Analysing residue names:
There are:   512      Water residues

  0 System              :  1536 atoms
  1 Water               :  1536 atoms
  2 SOL                 :  1536 atoms

 nr : group       !   'name' nr name   'splitch' nr    Enter: list groups
 'a': atom        &   'del' nr         'splitres' nr   'l': list residues
 't': atom type   |   'keep' nr        'splitat' nr    'h': help
 'r': residue         'res' nr         'chain' char
 "name": group        'case': case sensitive           'q': save and quit
 'ri': residue index

> a OW

Found 512 atoms with name OW

  3 OW                  :   512 atoms

> 2

Copied index group 2 'SOL'

  4 SOL                 :  1536 atoms
```

g_permute usage

Step3: Use g_permute to permute the trajectory

./g_permute -m 3 -f test_input.xtc -s tip3_water.gro -o permute_output.xtc -n index.ndx

```
Choose a group for the distance calculation:
Group     0 (          Water) has  1536 elements
Group     1 (             OW) has   512 elements
Select a group: 1
Selected 1: 'OW'

Choose the solvent group:
Group     0 (          Water) has  1536 elements
Group     1 (             OW) has   512 elements
Select a group: 0
```

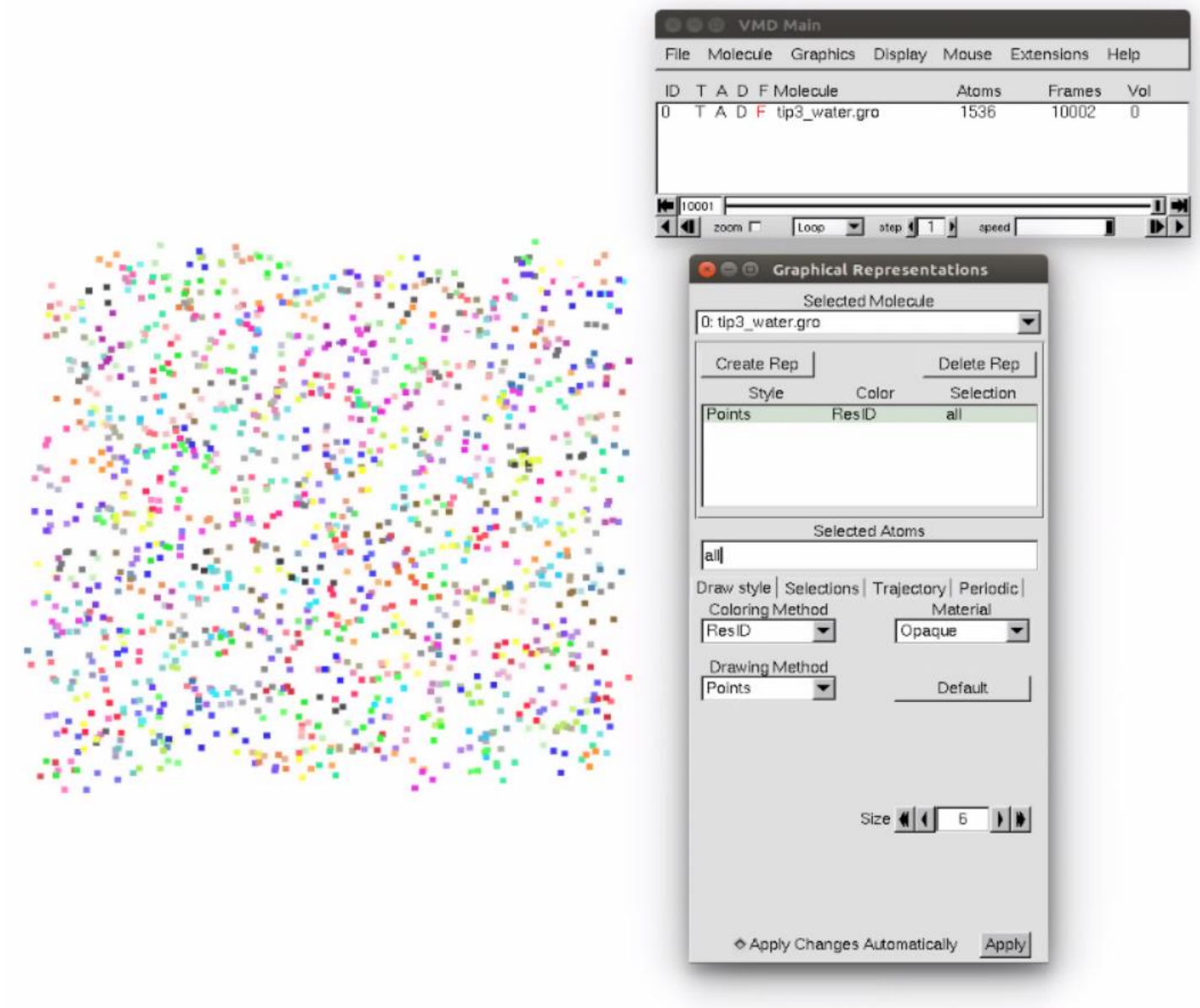
- This step will take from 5 – 20 minutes depending on the size of your trajectory and the number of atoms in your system.
- Once the step is complete, the output permute_output.xtc is generated.
- In order to check if your trajectory is permuted, use VMD to visualize the permuted trajectory.
- The procedure is given in the subsequent slides.

Visualization of unpermuted trajectory

- Load the raw unpermuted trajectory in VMD

- Represent the molecules as follows:
  Drawing Method -> Points
  Coloring method -> ResID

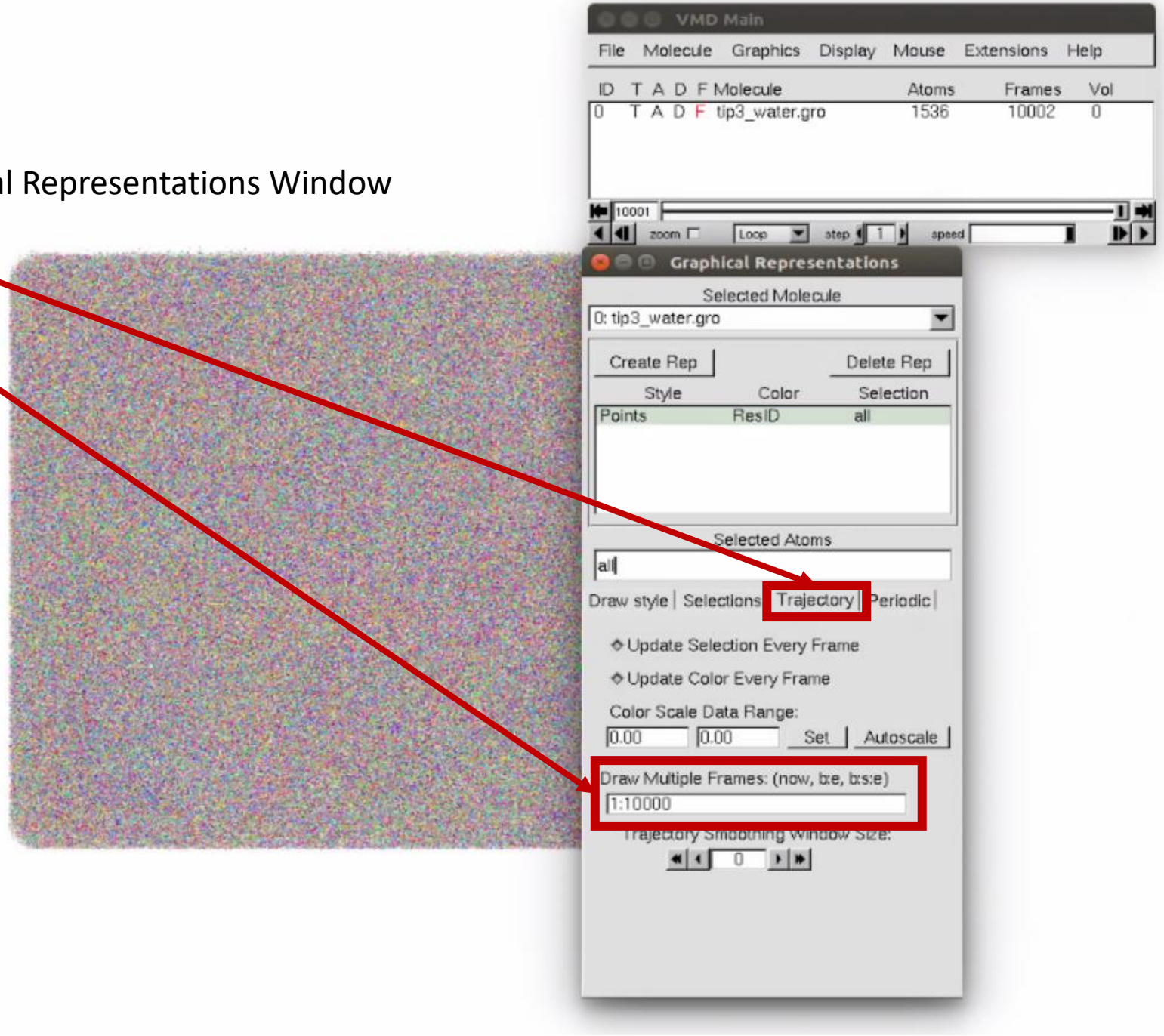
(For demo purpose, the points are displayed using size 6)



- Now go to Trajectory tab in the Graphical Representations Window

- Change the value in the
  'Draw Multiple Frames tab'
  1:1000 or 1:10000

- This value will plot the points from 1 to 1000 frames altogether in single window as shown in the figure.



Visualization of permuted trajectory

- Load the **permuted trajectory** and represent the water molecules in Points and drawing method as ‘ResID’.
- Now go to Trajectory tab in the Graphical Representations Window

- Change the value in the
  ‘Draw Multiple Frames tab’
  1:1000 or 1:10000

- This value will plot the points from 1 to 1000 frames altogether in single window as shown in the figure.

- Now each water molecule with a specific resID occupies specific volume instead of being dispersed all over the cuboid box.