

# Basic LAMMPS builds – LAMMPS Workshop 2017

Have a look at the LAMMPS website: <http://lammps.sandia.gov> Bookmark this page.

## WINDOWS INSTALL

Determine whether you are running 32-bit or 64-bit Windows

Click Start, then right click on “Computer” or “This PC” and select “Properties”

From the main LAMMPS Downloads page select **Windows installer package**, which leads to <http://rpm.lammps.org/windows.html>. Download the appropriate version for your system and select “Run”. You will be warned of an “Unknown Publisher”. Accept the default install path and the launcher will unpack the LAMMPS directories into your Program Files directory and add the directory to your path.

Now from the start corner, open Windows PowerShell, or CommandPrompt.

Type: `cd Documents`

Then: `xcopy /E 'C:\Program Files\LAMMPS 64-bit\Examples\' .\Examples`

## MACINTOSH INSTALL

Make sure that you have “Command Line Developer Tools” installed on your mac by typing “gcc” at a command prompt. If you don’t have them, then you will be prompted to install.

Follow the Linux Install instructions from there.

## LINUX INSTALL

From the LAMMPS Downloads page select “LAMMPS --- Stable version” and then “Download Now”

When the download completes, go to the directory containing the file and type “`tar -xvf lammps-31Mar17.tar`” This will unpack the LAMMPS directories.

Type “`cd lammps-31Mar17/src`”. Feel free to continue to explore with `ls` and `cd`. This is the source code of lammps and where the executable is configured and made using a unix/linux command called “make”

Type: `make mpi-stubs`

Then type: `make -j8 serial`

That last command did the work to compile the code and create an executable. This might take a while to complete. The machine type “serial” refers to the fact that this build is designed to run on only one processor and with the internal fft solver. When complete, the executable will `lmp_<machine>`, in this case `lmp_serial`

## Test LAMMPS on the obstacle example:

**LINUX/MAC:** In a Terminal window, move to `lammps/examples/crack`. Test that LAMMPS is setup. Type “`../src/lmp_serial -in in.crack`” You should get some output to the screen and a log file “`log.lammps`”.

**Windows:** Open a “Command Prompt” window. Change directory to `Documents/Examples/crack`

Type: `cd Examples\crack`

Then: `lmp_serial -in in.crack` to launch LAMMPS with the input script.

Then: `notepad in.crack` to edit the input script.

**Congratulations! You are ready for the hands-on activities...**

## LINUX/MAC Challenge: Setup LAMMPS for image dumps by installing libjpeg

**DOWNLOAD LIBRARY:** <http://libjpeg.sourceforge.net> Follow link to ijj.org to download the tarball for version 9b. Then extract the tarball, configure and compile/build it.

```
Type:      tar -xvzf jpegsrc.v9b.tar.gz
Then:      ./configure
Then:      make
Then:      sudo make install
```

### TELL LAMMPS TO USE IT:

In a Terminal window, move to lammps/src/MAKE. Edit the file Makefile.serial.

The following lines need to be changed:

```
LMP_INC    =  -DLAMMPS_GZIP  -DLAMMPS_MEMALIGN=64  -DLAMMPS_JPEG
```

Then point the code to the JPG library:

```
JPG_INC    =  <directory with jpeglib.h>
JPG_PATH   =  <directory with jpeg library>
JPG_LIB    =  -ljpeg
```

**REBUILD LAMMPS:** In lammps/src/ directory:

```
Type:      make clean-serial
Then:      make serial
```