**How To Set Up Dislocation Dynamics Code On A Local Machine**

*Note*: The commands in this tutorial are written for a tutorial system, so there may be slightly differences depending on your computer.

*If starting from Windows OS…*

1. Download and partition system to install Fedora (or other Linux operating system)
   * It is highly recommended that you backup your system prior to going through this process
   * May need to format a USB to be a disk drive to install Fedora – then reboot the computer in BIOS mode
   * <https://getfedora.org/>
   * <https://www.lifewire.com/guide-to-installing-fedora-linux-2202074> (how-to guide)
   * Partitioning roughly 100-200GB should be enough to start with

*If starting from Linux OS or after successfully partitioned system…*

1. Download and install Mercurial (hg) (<https://www.mercurial-scm.org/>)
   * Package will assist with source control management and branch updates
   * sudo dnf install hg
2. Download and install VTK and VTK developer tools (<https://vtk.org/>)
   * Package will provide visualization tools for DD
   * sudo dnf install vtk
   * sudo dnf install vtk-devel
3. Download latest gcc version
   * sudo dnf install gcc-c++
4. Download Model (DD) package using Mercurial (<https://bitbucket.org/model/model/wiki/Home>)
   * Use the terminal to move to the Documents folder (or wherever you want to place the Model code)
   * hg clone https://model@bitbucket.org/model/model Model <- uses hg to pull code and creates the Model folder
   * Then, cd into the created Model folder and enter the following command:
   * hg update DiscreteCrackMechanics <- moves user to the DiscreteCrackMechanics branch
     1. This is the most commonly updated branch by Giacomo
5. Download Eigen package using Mercurial
   * Library to assist with matrix math computation throughout the dislocation dynamics code
   * Use the terminal to move to the Documents folder (or wherever you want to place the Eigen library)
   * hg clone <https://bitbucket.org/eigen/eigen/> Eigen <- uses hg to pull code and creates the Eigen folder
   * OPTIONALLY, users can create symbolic links to the Eigen package with the terminal commands;
     1. sudo ln -s /home/cmcelfresh/Documents/Eigen Eigen
     2. sudo ln -s /home/cmcelfresh/Documents/Eigen/unsupported unsupported
     3. ALTERNATIVELY, you can edit the Makefiles in the Model code (to be discussed later)
6. Download and install FFmpeg package
   * FFmpeg package assists with creating videos from the produced DD figures
   * sudo dnf install ffmpeg
7. Download and install the TetGen package (<http://wias-berlin.de/software/tetgen/>)
   * TetGen assists with generating customizable meshes for DD simulations
   * Download online <http://wiasberlin.de/software/index.jsp?id=TetGen&lang=1#Download>
   * Follow [instructions](http://wias-berlin.de/software/tetgen/compile.html) on compiling TetGen code to create executable.
     1. Put all downloaded material in its own Tetgen folder
     2. Move into the folder using the command line
     3. make <- this should create a tetgen executable file
8. Download and install Intel Math Kernel Library
   * Find download at <https://software.intel.com/en-us/mkl/choose-download>
   * Select “Intel Parallel Studio XE”
   * Follow instructions for downloading for Linux as a student. You will be required to fill out a brief questionnaire about the usage of MKL for DD.
   * Once submitted it may take a day or so to received email confirmation of approval for the package. Download the package and follow the [Linux installation guide](https://software.intel.com/en-us/download/parallel-studio-xe-2019-install-guide-linux).
   * NOTE: During the installation make note of where the package is installed – its path will be necessary for updating the Model makefiles.
9. Update the Makefiles in the tutorial folders

*If starting from UCLA hoffman2…*

1. Download Model (DD) package using Mercurial (<https://bitbucket.org/model/model/wiki/Home>)
   * Use the terminal to move to the Documents folder (or wherever you want to place the Model code)
   * hg clone https://model@bitbucket.org/model/model Model <- uses hg to pull code and creates the Model folder
   * Then, cd into the created Model folder and enter the following command:
   * hg update DiscreteCrackMechanics <- moves user to the DiscreteCrackMechanics branch
     1. This is the most commonly updated branch by Giacomo
2. Download Eigen package using Mercurial
   * Library to assist with matrix math computation throughout the dislocation dynamics code
   * Use the terminal to move to a folder outside of the newly created Model folder (or wherever you want to place the Eigen library)
   * hg clone <https://bitbucket.org/eigen/eigen/> Eigen <- uses hg to pull code and creates the Eigen folder
   * OPTIONALLY, users can create symbolic links to the Eigen package with the terminal commands;
     1. sudo ln -s /home/cmcelfresh/Documents/Eigen Eigen
     2. sudo ln -s /home/cmcelfresh/Documents/Eigen/unsupported unsupported
     3. ALTERNATIVELY, you can edit the Makefiles in the Model code (to be discussed later)