Directory Structure for mloc

Eric Bergman
Global Seismological Services
December 8, 2015

This document describes the recommended arrangement of directories and files for use with the multiple event relocation program *mloc* v10.2.8 and later. The following schematic shows the main directory structure:

mloc
- **clusters**
- **documents**
- **mloc_build**
- **mloc_src**
- **mloc_working**
  - **tables**
    - **crust**
    - **ellipticity**
    - **faults**
    - **gmt**
      - **cpt**
      - **dem**
        - **ETOPO**
        - **GINA**
        - **GLOBE**
    - **kml**
    - **spread**
    - **stn**
    - **tau-p**
  - **utlities**

The paths that are underlined above are hard-wired in the main program (mloc.f90):

```
taup_path = 'tables/tau-p'
ellip_path = 'tables/ellipticity'
station_path = 'tables/stn'
cpt_path = 'tables/gmt/cpt'
dem_path = 'tables/gmt/dem'
psdre_path = 'tables/spread'
```

If you want to depart from the scheme shown above, you will need to edit mloc.f90 and recompile.

Top-Level Directories

The name and location of the top-level directory **mloc/** shown above is arbitrary. The names of the directories immediately below **mloc/** are also arbitrary. The role of each of these directories is described here:

clusters

The directories for individual earthquake clusters that will be analyzed using *mloc* are stored here. This directory can be located anywhere in the file system; it does not have to be under the top **mloc/** directory. It is convenient to have it nearby, however, because the directories that hold individual series of runs (see below) will be moved back and forth between here and the **mloc_working/** directory. Each cluster should be named after a geographic feature that exists within the cluster. Google Earth is a good tool for exploring possible names. Avoid using names that apply to a region larger than the cluster, (e.g., "zagros"), and choose a name that is fairly easy to type; you will be typing it frequently.

In practice, it is very common to have to make several (or many) series of runs on a cluster. The top directory for a set of such series can have a more descriptive name. Different series can be distinguished by significantly changed data sets (e.g., more or fewer events, new readings for some events) or application of a different relocation strategy. Each series would have its own directory under the main directory for the cluster, e.g.:

```
"Qeshm Island, Iran"
      qeshm1
      qeshm2
      …
      qeshm23
```

A specific run (relocation) within a series has its own index, e.g., qeshm23.1.

documents

This directory contains a variety of documentation about *mloc* and it's use. It can have any name you like and can be stored anywhere, but keeping it under the **mloc/** directory makes sense.

mloc_build

This directory is used by the makefile that controls compilation and building of the *mloc* executable. The makefile is stored here, and the object files for individual program units will be stored here also. The source code units to which the makefile refers are stored elsewhere, in the

directory **mloc_src/**. The executable file will be created here and then moved to the **mloc_working/** directory for use.

If you have more than one FORTRAN compiler that you wish to use, you can create several of these build directories and name the build file after the compiler, e.g., mloc_gfortran and mloc_intel. In that case you can name the executables accordingly, "mloc_g", "mloc_i", etc.

mloc_src

Source code files are stored here. The makefile in **mloc_bld/** must have the correct pathnames to the source codes. Having only the source code files in this directory makes editing easier, and is essential if more than one compiler is to be used.

mloc_working

This is where most of the relocation analysis takes place. The executable *mloc* is stored here. The absolute pathname of this directory must be supplied to *mloc* by the configuration file (mloc.conf), which must exist in this directory. The cluster-series directory (e.g., **qeshm23/**) of the cluster to be relocated must be moved to this directory from its permanent home in the **clusters/** directory.

The other essential directory that must exist in this directory is the **tables/** directory and it's subdirectories. It contains a variety of data files used by *mloc*, organized in a number of subdirectories, as shown in the schematic above. The contents of these directories are described below.

It is convenient, but not required, to create a directory **utilities/** in which to store the executables of several utility programs that are used regularly along with *mloc*:

lres: automatic flagging of readings with cluster residuals above a given threshhold
rstat: interactive investigation of cluster residuals for specific station-phases
xdat: automatic flagging of readings with gross outliers

To use these utilities, the executables are copied into the cluster-series directory, and then deleted when done, so it's convenient to have them close at hand.

tables/ and subdirectories

The **tables/** directory must exist within what I call the **mloc_working/** directory (or whatever you name it). Many pathnames in *mloc* are hardwired to look in the **tables/** directory for required data files.

crust

The **crust/** directory holds custom crustal models for use in *mloc* (command **lmod**). The argument to **lmod** is the pathname of the custom crustal model, relative to the *mloc* executable, so you may store crustal models in the **tables/crust/** directory or in the cluster-series directory or, indeed, anywhere you like. Therefore the **crust/** directory can be considered optional.

ellipticity

The **ellipticity/** directory holds a single data file (tau.table) that is used to calculated ellipticity corrections to theoretical travel times.

faults

The **faults/** directory holds digitized fault traces for plotting in GMT (command **fmap**). The argument to the **fmap** command is a relative pathname so the data file could be stored anywhere. The **faults/** directory can therefore be considered optional.

gmt

The **gmt/** directory contains two subdirectories, one (**cpt/**) to hold color palette tables and one (**dem/**) to hold digital elevation models for plotting topography. The **cpt/** directory should contain, at a minimum, the standard topo.cpt file that is distributed with GMT. If other color palette tables are available, a different one can be selected with the **cptf** command.

The **dem/** directory contains several subdirectories. Three of them are for the standard global DEMs distributed with *mloc*: GLOBE, GINA and ETOPO. The **dem1** command is used to select among these. If a high-resolution DEM is available for the particular source area of a cluster, it can be used for the baseplot and related plots that cover smaller areas, using the **dem2** command. The argument to **dem2** is a relative pathname, so it can be stored anywhere, within the **dem/** directory (or within a custom/ subdirectory), in the cluster-series directory or elsewhere in the filesystem.

kml

The **kml/** directory holds graphic files of icons used in the .kml file that is created for each run.

spread

The **spread/** directory holds one or more data files that prescribe default reading errors for specific phases. These are usually over-ridden, after a few runs, by the values determined in the HD analysis, but in some cases it may be useful not to use the empirically-determined reading errors.

stn

The **stn/** directory holds data files describing the locations of stations. The most important such file is the master station file, which is often adequate by itself to provide coordinates for all necessary stations. The name of the master station file (master_stn.dat) is hardwired as a default in the code, but an alternative file can be specified in the configuration file (mloc.conf). *mloc* also allows the use of multiple supplemental station files (command **sstn**) and these files can be stored in the **stn/** directory, in the cluster-series directory or elsewhere. This is also the normal place to store a data file of stations that are suspected of reporting bogus depth phases (command **bdps**).

tau-p

The **tau-p/** directory holds data files needed by the tau-p software to calculate theoretical travel times. Two files, a '.hed' and a '.tbl' file are needed for each earth model. *mloc* normally uses the model ak135, but the tau-p files for other 1-D earth models are also available and can be selected with the **taup** command in *mloc*.