**Purpose:**

This document describes how to create a metadata exchange format (mef) package for loading groups of metadata with accompanying files into GeoNetwork. There is sufficient detail required for the graphic overview and data files being added into the mef that it is best constructed programmatically.

Each metadata set in the mef package consists of a metadata record, optional graphics files, optional data files, and an “info.xml” document describing the package contents. It uses a custom python application (305-F05).

|  |  |  |
| --- | --- | --- |
| **Step** | **Major Activity** | **References, Forms and Details** |
| 1 | Open a python shell | ActivePython is one example of a suitable shell. It has a graphical user interface |
| 2 | Navigate to the python application   * File – open - … | Application is listed in 307-F03 |
| 3 | Modify the paths, following the expected folder structure mentioned in processes 307-01 and 307-02. |  |
| 4 | Modify the site name convention and the site UUID |  |
| 5 | Run the application. Outputs are:   * real time feedback on-screen * each metadata package will be formed in tmp, consecutively * a log file and the final mef package (called <something>.zip) will be placed in the mef folder |  |
| 6 | Errors may be noted on-screen or in the log:   * path errors * missing thumbnails when expected * missing data when expected   Identify and correct |  |
| 7 | Ensure the resulting mef is less than 100 MB   * if not, split into one or more packages using the partial results from the tmp folder | The 100MB limit can be changed, but it is probably a reasonable size for upload. |
| 8 | Upload MEF to GeoNetwork | Follow process 305-06 |