Reply to Riccardo Zaccarelli's comments (RC2) on "obspyDMT: A Python Toolbox for Retrieving and Processing of Large Seismological Datasets"

---

1) On section 3.8 (line - or paragraph - 25) the sentence "Although this file is fully customizable" sounds a bit misleading or needs clarification:
- If the authors mean that the processing steps can be achieved by flags and - for more experts users - the python file in the package is always editable, I would remove or rephrase the sentence for two reasons: first, modifying open source code (let alone licensing for the moment) is a feature common to any open source software and I do not see why it is worth a particular mention. Second, it might be interpreted like the authors encouraged the modification of their own source code, which sounds quite odd (and exposes any user to potentially breaking the code, at their own risk).
- On the other hand, if a particular customization has been implemented, maybe the sentence might deserve a little bit more explanation

REPLY: Thank you, clearly this sentence was too short and cryptic. We are not talking about changing source code, as hopefully our new wording makes clear. Rather we are enabling users to customize their waveform processing via scripting calls to external programs such as obspy and SAC. Virtually any processing tool which can be called from the command line (e.g., obspy, SAC or a combination of these tools) can be integrated, by writing a processing script **my_proc_unit** and executing is via the obspyDMT call **--pre_process my_proc_unit**. Currently two example scripts are also provided, located at /path/to/my/obspyDMT/obspyDMT.

OLD:
Although this file is fully customizable, several common processing steps can be done via options flags (without changing/writing new processing instructions).

NEW:
This scripting file can be freely edited by the user and may include calls to external waveform processing programs such as obspy or SAC. This vastly expanding the possibilities for waveform processing and lets users easily adapt and integrate functionality from earlier, non-obspyDMT work flows. Instructions in this file are written at the waveform level, and obspyDMT applies them to all waveforms in the entire data set (in serial or in parallel mode). The default file included in the current distribution, /path/to/my/obspyDMT/obspyDMT/process_unit.py, can perform routine processing steps such as resampling, data format conversion, and instrument correction. These steps can be accessed via dedicated option flags, each of which results in the execution of only the appropriate part of processing script process_unit.py (see --pre_process option flag). Hence a user requiring only these routine operations need not create or modify a processing script file.