

# OpenCPS experience in my summer seismic preprocessing workshop

*Stewart A. Levin*

## ABSTRACT

Many students (and faculty) throughout the Earth Sciences have at least an occasional need to deal with seismic data. To help them handle some basic data import, preparation and, as is common here at Stanford, export, I ran a seven session workshop over the summer. During the course of the workshop, I exposed students to multiple packages: Seismic Unix, ProMAX/SeisSpace, and, the very recently arrived, OpenCPS. In this report, I discuss our experience with that last package, both as a vehicle for allowing students to easily focus on the geophysics and for the capabilities it provides the casual user.

## INTRODUCTION

Ever since I returned to Stanford two years ago, I have been helping students and faculty import and prepare seismic data for use in their research. With many students around over the summer and not tied up with classes, I decided to offer a summer hands-on workshop (Fig. 1) for those who might be interested. About ten students showed up for some or all of the sessions, originating from SEP, borehole seismology, crustal seismology and even petroleum (um Energy Resource) engineering.

During the course of planning and preparation, one of Simon Klemperer's students informed me that he had recently obtained a license for Open Geophysical Inc.'s OpenCPS package and had found it easy to use and very helpful in understanding a seismic dataset he had recently recorded in the field. This package was designed and developed by a group headed up by Roland Gunther, a former Stanford student who had done undergraduate work with Simon Klemperer and graduate work at the Stanford Exploration Project. We were able to transfer the license to a machine that the students could all access, allowing me to use the software as part of the workshop exercises and instructions.

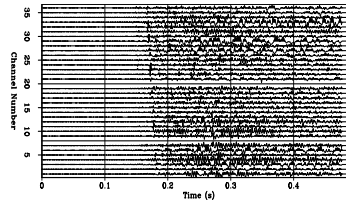
As detailed writeups of the sessions are available from my SEP website under [sep.stanford.edu/oldsep/stew/Summer2013Workshop/](http://sep.stanford.edu/oldsep/stew/Summer2013Workshop/), I will limit my discussion to the highlights or lowlights of that experience and refer the interested reader to my website for fuller details.

## Summer Workshop on Seismic Data Preprocessing

Instructor: Stewart A. Levin

Where: Mitchell A-65

When: Tuesdays 10 AM



Purpose: Assist students and faculty with decoding and preprocessing of seismic datasets, including exposure to various software tools available to the School of Earth Sciences.

Organizational Meeting: June 25<sup>th</sup>

Flexible syllabus includes:

- Basic terms and units (sample rate, stacking, coordinate reference system, velocity, ...)
- U CN DCD SSMC DAT (how to dump/scan and inspect binary seismic formats)
- Geometry (importing and applying auxiliary files)
- Visualization
- Filters and Deconvolution
- Velocity Analysis
- Stacking and Signal Enhancement

Figure 1: Workshop announcement

## OPENCPS

The “CPS” in OpenCPS refers to the former ConocoPhillips seismic processing system subsequently converted to an open-source version at [sourceforge.net/projects/cpseis/](http://sourceforge.net/projects/cpseis/). These modules formed the initial basis for the core geophysical methods of OpenCPS, though the user parameter interface to them has, thankfully, been almost totally replaced by Open Geophysical, which built its user interface framework on top of the Eclipse<sup>1</sup> open source development and application framework.

### Installation

Installation of OpenCPS requires two things: a compressed tar file and a license file. The compressed tar is downloaded from the [opengeophysical.com](http://opengeophysical.com) website and expanded into any chosen installation directory. The website download is updated quite frequently to include any new modules and bug fixes they’ve vetted. At our site, the system administrator created a symbolic link from `/usr/local/bin/opencps` to the startup script of that same name in the installation directory. After placing the license file into the installation directory, no further configuration is required to invoke the `opencps` command, although supplemental batch queue configuration would be useful for large scale processing such as migration of 3D prestack marine data.

### Initialization

When the user interface first comes up, there are a few preferences and defaults that can usefully be set up, the most important being the default location for where seismic projects should be created. Since those projects will contain volumes of seismic data, the initial choice of the user’s home directory should be overridden. (When a new project is created, you can, of course, override the default.) In my summer workshop, several students initially got stuck because of storage quota limits on their home directories. I got stuck later on because the package sometimes creates large temporary work files under `$HOME/.eclipse`. (That was fixed by moving `.eclipse` elsewhere and making a symbolic link from my home directory to the new spot.)

### Overall Impressions

OpenCPS advertises itself as *Interactive Seismic Processing* and we all found it quite easy to set up a project, use the canned flow wizards to quickly run through most basic preprocessing steps, and view the results with just a right click on the relevant seismic dataset or table.

---

<sup>1</sup>[www.eclipse.org](http://www.eclipse.org)

When we ran into problems, mostly the result of our academic system configuration, the Open Geophysical developers were quick to address them, either making quick bug fixes or giving us workarounds. They were also very responsive to suggestions for clarifying user interface verbiage that confused us novices.

The development environment was sufficient for us to write a deconvolution tool or two with only minor glitches in the installation process. (You clone one of their examples, tell the user interface where your code and libraries reside, and hit Refresh on the Tools list.)

The OpenCPS system wears two hats—Batch and Interactive. The difference is that Batch runs a flow and writes output, Interactive displays data instead of writing output and repeatedly runs the flow in response to changes in flow tool parameters. In addition, the Interactive view provides quick ways to bounce around through even a massive dataset, allowing you to fine tune parameters and tables before running the whole input through the flow. I found this quite useful in playing around with the parameters in my port of Jon’s Ricker-compliant decon. I haven’t exercised the parallel capabilities of the Batch view, in particular, the cluster job replication template mechanism, so I cannot comment on that area. I do note, though, that OpenCPS definitely takes advantage of multiple cores to very nicely utilize available I/O bandwidth.

One area I found annoying was seismic data gain control. All too often when I switched the display parameters (exposed by clicking on an arrow at the top right of the plot window) from even something simple like smooth interpolation to nearest neighbor interpolation, I had to manually regain by as much as a factor of 5.

Another negative is the limited number of tools, presumably because their open source business model incorporates upselling more sophisticated modules, and the buggy/limited interface to the Seismic Unix tool list they provide. For us these are not really an issue for the main purpose of importing and cleaning up data before export to SEG-Y, SEP, or, perhaps, Matlab.

## CONCLUSIONS

Overall, OpenCPS works quite nicely to provide the limited list of capabilities and the flexible interactivity that most users here might desire. The folks who have been using ProMAX (but not SeisSpace, I rush to clarify) found their work moving along more rapidly under OpenCPS. It still has some room for production hardening, but overall, it nicely fills the basic preprocessing niche that I targeted in my workshop.