

Electronic supplement

Introducing novices to active-source seismology via SeismicUnixGui

Juan M. Lorenzo, Michael Hubenthal, Nathan W. Benton, Emilio E. Vera

- Design strategy for SeismicUnixGui (SUG)
- A Data Set and Tutorial
- Figure S1 An image of data from three, raw, muted shotpoint gathers corresponding to Step 8 (Flow 5D) in the online tutorial (see Data and Resources in the main text).
- References

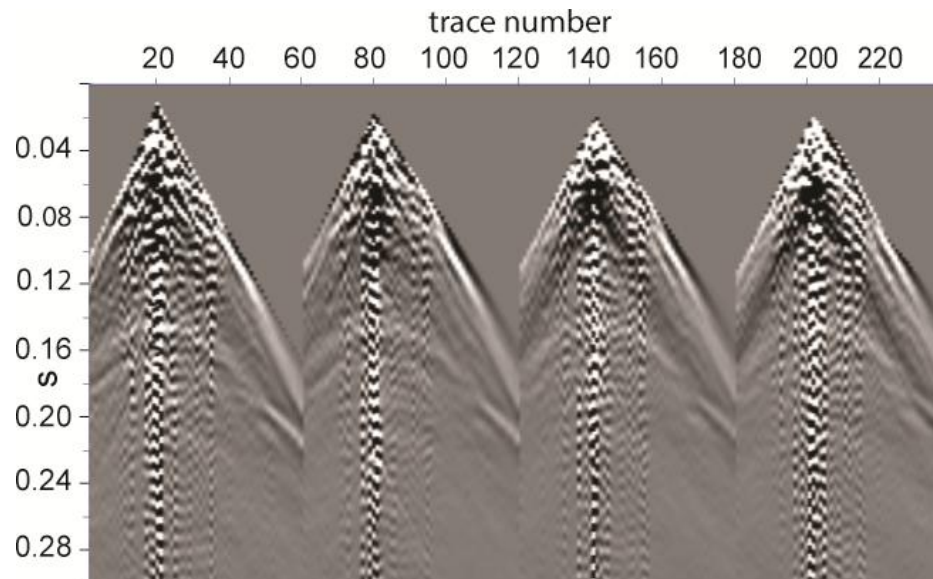
12

13 **Design Strategy for SeismicUnixGui (SUG)**

14 SUG (Figure 2, in main body of text) is built with Perl/Tk (Simmons and Rézic, 2024), a mature,
15 well-documented and free Perl-based extension of the Tk GUI toolkit (Ousterhout, 1993)
16 originally developed for use with the X Window System (Scheifler and Gettys, 1986; X.org, 2024)
17 on Unix (Kernighan and Mashey, 1979). For reference, Tkinter (Lundh, 1999), which is the default
18 graphical user interface for Python (2024), is also based on the same toolkit. SUG development
19 blends an early procedural programming style that evolved through extensive refactoring toward
20 an object-oriented programming approach. Perl/Tk organization lends itself to the use of a
21 specialized extension to Perl, known as Moose (Etheridge, 2024), which facilitates an object-
22 oriented style in programming.

23 In SUG, application of the wrapper scripts does not modify the original SU code, and the user
24 is unaware of the repeated background loop calculations that ensue. New options added by
25 wrappers can be ignored by the user or readily removed in later updates to SU modules. Wrappers
26 can help introduce intuitive and easier-to-understand names for variables that more directly express
27 their functionality. For example, instead using of a non-descriptive variable name such as “b” to
28 determine the maximum amplitude for display in a plot, the user is provided with a more self-
29 explanatory name such as “high-clip”, which is intuitively more obvious to the user. In addition,
30 to aid beginning students, the classical SU help manual for each module can be made to appear
31 conveniently in a separate window with a click of the third mouse button. For comparison, SWIG
32 (2024), a popular wrapper software development tool that allows reuse of programs written in C
33 and C++, builds its wrappers using C code.

34 A Data Set and Tutorial



35 **Figure S1. An image of data from three, raw, muted shotpoint gathers, corresponding to Step**
36 **8 (Flow 5D) in the online tutorial (see Data and Resources in the main text).**

38 References

- 39 Etheridge, K. (2024). Moose, postmodern object system for Perl 5,
40 <https://metacpan.org/pod/Moose>, last accessed October 23, 2224.
- 41 Kernighan, B. W., and J. R. Mashey. (1979). The UNIX™ programming environment, Software:
42 Practice and Experience, 9, 1-15.
- 43 Lundh, F. (1999). An introduction to tkinter, accessed October 8, 2024 at
44 <https://web.archive.org/web/20130512143826/http://www.pythonware.com/library/tkinter/introduction/index.htm>
45

46 Ousterhout, J. K. (1993). Tcl and the Tk Toolkit, Addison-Wesley Professional Computing
47 Series, Addison-Wesley Publishing, Reading, MA Python, (2024) <https://www.python.org/>,
48 last accessed October 23, 2024.

49 Scheifler, R. W., and J. Gettys (1986). The X window system, ACM Transactions on Graphics 5,
50 79-109.

51 Simmons, N.-I., and S. Rézic (2024). Tk- A graphical user interface toolkit for Perl., Available
52 from <https://metacpan.org/dist/Tk/view/Tk.pod> (Version 804.036., last accessed October
53 2024)

54 SWIG, (2024). Simplified Wrapper and Interface Generator, <https://www.swig.org/>, last accessed
55 October 24, 2024.

56 X.org, (2024), <https://www.x.org/wiki/XorgFoundation/>, last accessed, October 23, 2024.