Submission code: 6AZNJBKZAQ

See this and other abstracts on our website: https://home.sbgf.org.br/Pages/resumos.php

BotoSeis-web: An interactive web application for seismic data processing using Seismic Unix

Jorbralyson Freire de Oliveira Ribeiro (BTI/IMD/UFRN), Carlos Eduardo Dos Santos Garabito (BTI/IMD/UFRN), Pedro Miguel Varela Costa (BTI/IMD/UFRN), German Garabito (DPET/CT/UFRN)

BotoSeis-web: An interactive web application for seismic data processing using Seismic Unix

Please, do not insert author names in your submission PDF file

Abstract

There are several free and open-source software for seismic data modeling, processing and inversion. Seismic Unix (SU) is one such software that contains seismic data modelling and processing tools for exploration geophysics and seismology, widely used in several institutions for research and educational activities. SU has been designed for utilization via the command line of a Linux terminal, so its use in teaching activities can present certain challenges. BotoSeis has been developed to facilitate the use of SU processing techniques by means of an interactive graphical interface that allows the user to create, edit and execute seismic data processing flows. However, its use is still limited to the Linux operating system and requires the installation of SU, which can still be challenging for some users. This work presents the BotoSeis-web software, which aims to further facilitate the use of the SU programs from a web browser. This new software is platform-independent, meaning it can be accessed on any device with an internet connection, allowing the software to reach more users. The BotoSeis-web has a main interactive web interface for creating and submitting seismic data processing flows, an API (Application Programming Interface) responsible for processing seismic data and interacting with the user via the main web interface, and a data-viewer for analyzing input and output seismic data in a browser. Here, the first version of the BotoSeis-web software is presented, the software architecture is described, and the main and data visualization interfaces are shown.

Introduction

Seismic data processing comprises a series of mathematical and computational processes applied to transform recorded seismic data into images that show the Earth's subsurface structures. Seismic datasets are typically processed using specialized commercial software, which offers interactive and user-friendly interfaces for ease of use. These software solutions are expensive, and although they provide academic licenses, their availability is limited to processing centers and laboratories of educational institutions.

There are many free and open-source software solutions for processing seismic data, the most popular being Seismic Unix (Cohen & Stockwell, 2008) and Madagascar (Fomel et al., 2013), both of which are used by several educational and research institutions around the world. The Seismic Unix (SU) software was originally developed for use on computers running Linux operating system, where processes are executed via command lines in a Linux terminal. This way of using it, with long sequences of commands, makes teaching and learning seismic processing difficult. In addition, installing and using this software on personal computers requires a certain level of proficiency in Linux operating systems, commands and shell programming, which may represent a challenge for some users.

To facilitate the use of the SU software in teaching seismic processing, the BotoSeis software was introduced (Garabito et al, 2012; Garabito, 2019). BotoSeis is desktop software, developed in the Java programming language, which has an interactive main window that allows the user to build and execute seismic data processing flowcharts using the SU programs, and additionally has tools for data visualization and velocity analysis. However, installing Seismic Unix and BotoSeis requires a certain degree of familiarity with the Linux operating system, which is also a limiting factor for using both.

To overcome the challenges encountered by the Seismic Unix and BotoSeis software, this work aims to develop new interactive software for processing seismic data from a web browser. This web-based application, called BotoSeis-web, will have an intuitive user interface and will allow the user to focus on learning and applying seismic processing techniques. BotoSeis-web can be accessed from any device with an Internet connection, eliminating the need for specific software installations on each user's device. This paper details the structure and development tools of the BotoSeis-web software, which comprises three main components: The Application Programming Interface (API): A set of definitions and protocols responsible for executing seismic data processing and handling user interaction via HTTP requests. The main web client (web application): An interactive web interface that allows users to build and submit processing flows to be executed by the API. The data-viewer: A web tool that allows users to interactively view and analyze seismic input and output data. We will also present the initial version of BotoSeis-web, highlighting its main working interface and the data-viewer.

Method

The BotoSeis-web architecture is shown in Figure 1 and was modeled using the C4 notation (Brown, S., 2019). This architecture is divided into three main components: Web app, API, and the data-viewer. These components communicate through the API container. There is also an administration tool for managing seismic processing programs, allowing users to add new programs or update existing ones.

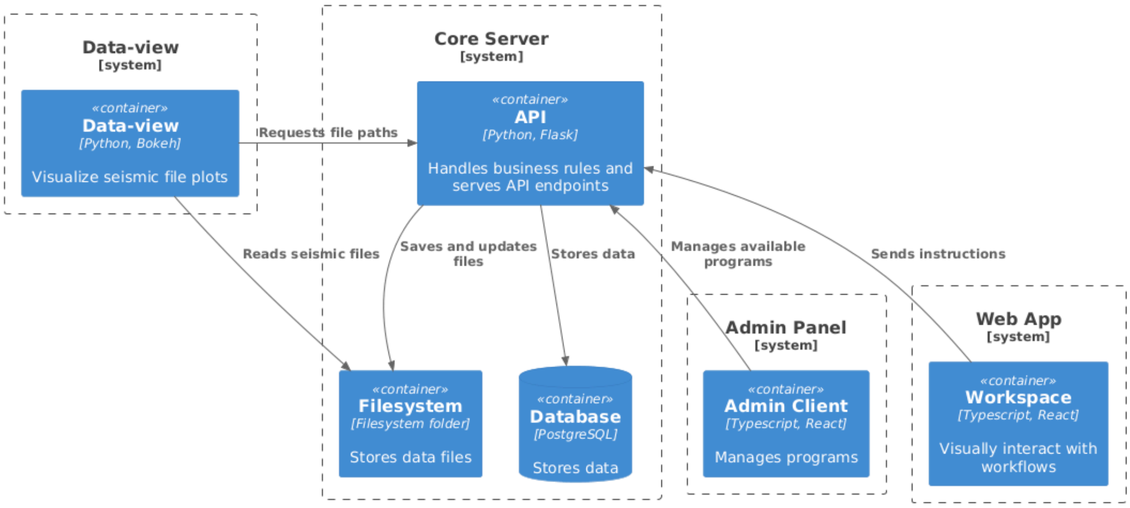

Figure 1: Botoseis-Web container-level architecture

The webapp was developed with TypeScript (Bierman et al., 2014) and the React framework (Meta, 2022). A centralized state machine was used to handle user interactions while waiting for HTTP responses in the background. Types and error handling make the state machine reliable.

The API was developed with Python using the Flask framework, chosen for its lightweight and flexibility. A PostgreSQL database was modeled using the ORM SQLAlchemy to represent key entities and their relationship. The code base follows a layered architecture design pattern to have modules responsible for each aspect of the application. This relational model is a pattern that aims to keep a clear organization of processing tasks and persistent tracking of user progress. User authentication and session security are based on JSON Web Tokens (JWT) for access control and Bcrypt for password hashing. JWT is stored in cookies, so keeping the token securely is easier when navigating between clients from the webapp to the data-viewer, avoiding making the user log in twice. The API constructs a string formatted according to Unix shell syntax, incorporating I/O redirection and pipelines. The resulting string is executed via subprocess calls.

The data-viewer tool was developed with Bokeh (Bokeh Development Team, 2025), which is a Python library for creating interactive visualizations in web browsers. To support the data-viewer, "seismic-io" was developed, a Python library tailored for reading and writing seismic data in the Seismic Unix format (Cohen & Stockwell, 2008). Using this library, we also implemented the "perc" and "gain AGC" functions, particularly useful for data visualization. This approach enabled the implementation of the following features: opening stacked data, opening subsets of gathers from a larger dataset, and different visualization modes such as wiggle trace, variable area wiggle trace, and variable density. Additionally, standard visualization operations like zooming, panning, and saving plots are available, powered by the Bokeh library.

Results

The main window of the BotoSeis-web is shown in Figure 2. This workspace organizes projects by dividing the workflows into lines, where each workflow allows the user to store an individual list of commands, where it is possible to insert new commands from the already registered programs, remove commands, re-order commands, and update their parameters. When the workflow is ready, it can be sent to the API to execute its instructions, receive the process logs, and display the processing output on the data-viewer.

When a workflow is successfully run, it creates a dataset. A dataset stores the process resulting file and the setup used to achieve this result, including the input file, the ordered commands, and their parameters. The input files can be individually displayed on the data-viewer.

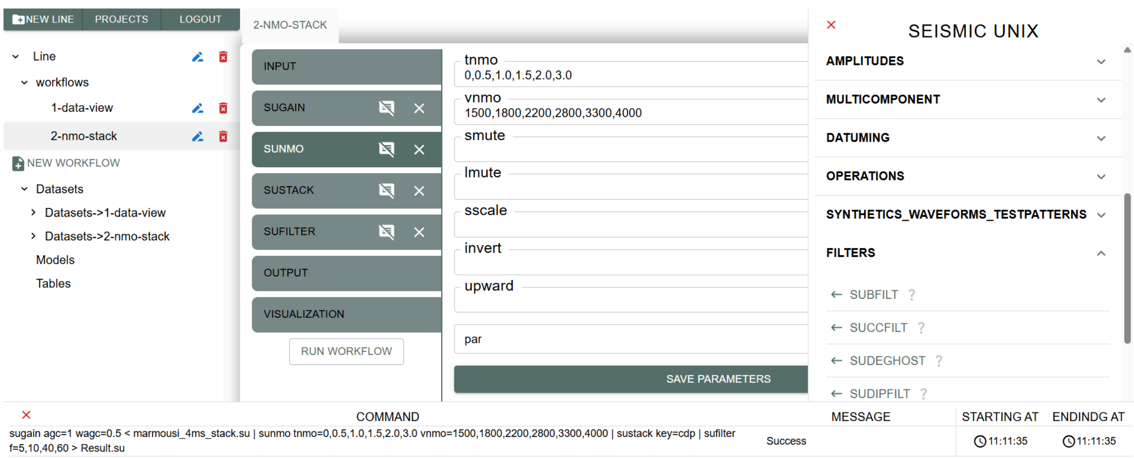

Figure 2: Botoseis-Web workspace showing a processing flow to apply various processes. The output data is saved in a file and is also displayed in the data-viewer, as shown in Figure 3.

The result of BotoSeis-data-viewer is presented in Figure 3, which shows the stacked image of the Marmousi (Bourgeois, et al. 1990) data using the 'variable density' visualization mode with a grey color map. Two processes have been applied to the data to improve visualization: a percentile clip of 97 and an AGC gain with a 0.5 second window. Standard operations such as zooming, panning, and downloading the graph as an image are also available.

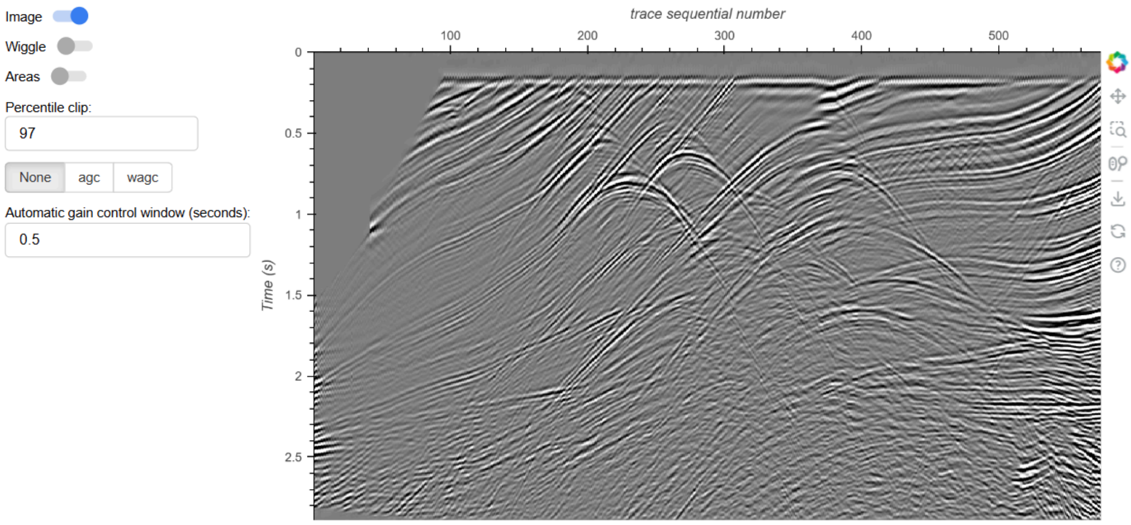

Figure 3: Botoseis-web data-viewer: Stacked section of Marmousi dataset.

Conclusions

This work presents the first version of Botoseis-web, a web-based platform for seismic data processing and visualization using the Seismic Unix package. Its key advantage over similar software is its web-based nature, making it ideal for teaching and self-learning learning activities.

Acknowledgments

This work was supported by PIBIC-UFRN and CNPq through research project 306393/2022-0.

References

Bokeh Development Team, 2025. Bokeh: Python library for interactive visualization, https://bokeh.org

Brown, S., 2019, Software Architecture for Developers (2nd ed.) Leanpub

Cohen, J. K. and Stockwell, Jr. J. W., 2008, CWP/SU: Seismic Unix Release No.34: an open-source software package for seismic research and processing, Center for Wave Phenomena, Colorado School of Mines, Colorado, USA.

Bourgeois, A., Bourget, M., Lailly, P., Poulet, M., Ricarte, P., Versteeg, R. (1990). "Marmousi, model and data". EAEG Workshop - Practical Aspects of Seismic Data Inversion. European Association of Geoscientists & Engineers.

Garabito, G., Lima W., Almeida, G., and Cruz, J. C.R., 2012, BotoSeis: An Interactive Interface for Seismic Processing with Seismic Unix, 74th EAGE Conference & Exhibition incorporating, Expanded Abstracts, doi: 10.3997/2214-4609.20149873.

Garabito, G., 2019, Tips for seismic data processing with BotoSeis & Seismic Unix, https://germangarabito.wordpress.com

Fomel, S., Sava, P., Vlad, I., Liu, Y., and Bashkardin, V., 2013, Madagascar: open-source software project for multidimensional data analysis and reproducible computational experiments: Journal of open research software, 1, No. 1, 1–4.

Bierman, G., M. Abadi, and M. Torgersen, 2014, Understanding TypeScript: In European Conference on Object-Oriented Programming, 257–281.

Meta, 2022. React: A JavaScript library for building user interfaces (Version 18). https://reactjs.org/