An unfinished tutorial for



L\_SU,

a graphical user interface

for Seismic Unix (CSM),

under Linux

Louisiana State University

Dept. Geology and Geophysics

Baton Rouge

Juan M. Lorenzo

Version: 0.3.9 (November, 2019)

[2 General Information 2-4](#_Toc26616986)

[2.1 Acknowledgements 2-4](#_Toc26616987)

[2.2 What is L\_SU? 2-4](#_Toc26616988)

[2.3 GUI Sections 2-6](#_Toc26616989)

[2.3.1 Overview 2-6](#_Toc26616990)

[2.3.2 Top Menu 2-6](#_Toc26616991)

[2.3.3 Side Menu 2-7](#_Toc26616992)

[2.4 What is an example directory structure for a Project? 2-8](#_Toc26616993)

[2.4.1 Where are my data sets stored? 2-8](#_Toc26616994)

[2.4.2 Copying data into the project directory structure from elsewhere in the system 2-8](#_Toc26616995)

[2.4.3 Where are my flows kept? 2-9](#_Toc26616996)

[2.5 Text conventions in this tutorial and their meaning 2-9](#_Toc26616997)

[2.6 Glossary 2-9](#_Toc26616998)

[3 Quick start: New Projects, Old Projects and Demos 3-11](#_Toc26616999)

[3.1 Create your first project (IRIS demonstration data set) 3-11](#_Toc26617000)

[3.1.1 Install example data sets 3-11](#_Toc26617001)

[3.1.2 Create a new project called Servilleta\_demos 3-11](#_Toc26617002)

[3.1.3 For the IRIS Data set, confirm you are working Project called “Servilleta\_demos” 3-13](#_Toc26617003)

[3.1.4 For the GPR data sets (from LSBB), confirm you are working Project called “LSBB” 3-14](#_Toc26617004)

[3.2 Open a pre-existing project 3-14](#_Toc26617005)

[3.2.1 The following instruction starts the program, and open the pane of the Project Selector window: 3-14](#_Toc26617006)

[3.3 Running your first flows 3-15](#_Toc26617007)

[3.3.1 Perl and Shell script flows generated by L\_SU 3-17](#_Toc26617008)

[3.3.2 Access to Documentation 3-18](#_Toc26617009)

[4 Simple Processing Flow: IRIS Data Set, Socorro New Mexico 4-19](#_Toc26617010)

[4.1 Processing steps 4-19](#_Toc26617011)

[4.2 STEP 1. File format conversion 4-20](#_Toc26617012)

[4.3 STEP 2. Concatenate files 4-21](#_Toc26617013)

[4.4 STEP 3. Clean headers 4-21](#_Toc26617014)

[4.5 STEP 4. Window the shotpoint gathers 4-22](#_Toc26617015)

[4.6 STEP 5. Negative stack 4-22](#_Toc26617016)

[4.7 STEP 6. Modify header files--gx,ep,sx 4-22](#_Toc26617017)

[4.8 STEP 7. Modify Header files--offsets 4-23](#_Toc26617018)

[4.9 STEP 8. Modify Header files--Make CMP's 4-25](#_Toc26617019)

[4.10 STEP 9. Dip filter 4-25](#_Toc26617020)

[4.11 STEP 10. Test Muting of surface waves and refracted waves 4-27](#_Toc26617021)

[4.12 STEP 11. Test Semblance Analysis 4-29](#_Toc26617022)

[4.13 STEP 12. Normal Moveout and Stacking and Migration 4-31](#_Toc26617023)

[5 Simple Processing Flow for GPR data 5-32](#_Toc26617024)

[6 Perl and Shell script flows generated by L\_SU 6-33](#_Toc26617025)

[6.1 IRIS Data Set, Socorro, New Mexico 6-33](#_Toc26617026)

[6.2 GPR data 6-36](#_Toc26617027)

## 

# General Information

## Acknowledgements

This project is possible only because of the selfless work of others. I have shamelessly copied and modified notes extensively from the Colorado School of Mines website (Stockwell) for S\*nix. Over the years, many students have also contributed to these notes: Class of 2008: Erin Walden, Kody Kramer, Erin Elliott, Andrew Harrison, Andrew Sampson, Ana Felix, JohnD’Aquin, Russell Crouch, Michael Massengale, and David Smolkin; Chang Liu (2013), Nevra Bulut (2019).

I will greatly appreciate any and all questions you have regarding installation and running of any of the programs to help us continue developing L\_SU. Pelae send your questions to [gllore@lsu.edu](mailto:gllore@lsu.edu). Please indicate what your operating system is and whether you have administrative priviliges (preferred).

Thanks,

Juan Lorenzo, BatonRouge, Dec. 5, 2019

## What is L\_SU?

L\_SU, a graphical user interface (GUI), serves to select and build sequences of Perl modules and their parameters. L\_SU generates two versions of these instructions in text files. These text files contain a shell and a Perl script version that can be modified and also executed independently of this GUI and from the command line.

Seismic Unix (Stockwell, 1999) is a widely distributed free software package for processing seismic reflection and signal processing.  In Seismic Unix, a sequence of independent programs receive modify and generate data files of streams of data that are displayed on the screen. The data file is read in and the generated output data are handled internally by stdin, stdout functions in C while the data exchanges between programs and the linux operating system are managed from the command line via pipes “|” and redirections “> or <” respectively. Traditionally, the instructions on the command line can be assembled and saved as re-usable bash scripts. L\_SU assembles these same scripts for the operating system to run with the help of modules written in Perl. L\_SU generates these scripts within the directory of the user and thes scripts can be run independently of L\_SU running.

L\_SU is written using Perl/Tk which is mature, well-documented Perl module that allows its users to construct graphical user interfaces.

In a classroom environment, shell scripting of SU modules engages students and helps focus on the theoretical limitations and strengths of signal processing.  However, complex interactive processing stages, e.g., selection of optimal stacking velocities, killing bad data traces, or spectral analysis requires advanced flows beyond the scope of introductory classes.  In a research setting, special functionality from other free seismic processing software such as SioSeis (UCSD-NSF) can be incorporated readily via an object-oriented style to programming.

An object-oriented approach is a first step toward efficient extensible programming of multi-step processes, and a simple GUI simplifies parameter selection and decision making.  Currently, in L\_SU, Perl 5 packages wrap 19 of the most common SU modules that are used in teaching undergraduate and first-year graduate student classes (e.g., filtering, display, velocity analysis and stacking).  Perl packages (classes) can advantageously add new functionality around each module and clarify parameter names for easier usage.  For example, through the use of methods, packages can isolate the user from repetitive control structures, as well as replace the names of abbreviated parameters with self-describing names.  Moose, an extension of the Perl 5 object system, greatly facilitates an object-oriented style.  Perl wrappers are self-documenting via Perl programming document markup language.

An automatic directory structure is created for the user in which data and programs are distributed according to a pre-defined hierarchy. All the directories and minimal files needed by L\_SU are created whenever a new 'Project' is created within the 'Project Selector' tool.  The user can also create new projects within main GUI of L\_SU as well as selecting different projects.  At all times the user can use linux commands to navigate freely through the directories. Sometimes the user may find it convenient to create new subdirectories within the existing file structure. L\_SU will not be able to detect these folders and their contents.

## GUI Sections

A screenshot of a social media post

Description automatically generated

### Overview

The main GUI is divided into 4 areas: Top Menu, Left Side Menu, Parameter Names and their Values, Four flow boxes, and a Message area. The large cross (**X**) in the top-left corner is used to kill many unwanted graphical process running in the background.

### Top Menu

There are more than 400 independent programs are available from Seismic Unix. Currently L\_SU implements over 65 of these.

### Side Menu

#### Tools

**Project**: Defines the directory structure for data sets and programs in many languages, e.g. matlab, R, Perl etc.

A screenshot of a cell phone

Description automatically generated

**Sseg2su**: Converts SEG-2 formatted data into the su format which is a simplified SEG-Y format.

% Sseg2su

**Sucat**: Concatenates multiple files of any format into a single file. These files can have names related by a continuous sequence of integers, e.g., Seismic Unix data files: 1000.su, 1001.su, 1002.su. If not, a list of names can be specified. Output files from interactive muting or velocity analysis and that have specific “par” formats can be handled.

% Sucat

## What is an example directory structure for a Project?

### Where are my data sets stored?

Before starting a new project you should understand the file structure in which programs and data sets are stored. The main directories are shown above.

### Copying data into the project directory structure from elsewhere in the system

If you want to copy seismic data already in su (seismic unix) format copy it with the following instruction, but first move yourself into the directory that receives the data.

Example 1:

% cd PROJECT\_HOME/seismics/**data**/site/component/line/username

Example 2:

% cd /home/gllore/seismics/data/Servilleta/H/1/gllore

Example 2:

% cp data /home/refseis18/Aug27\_lab1/\*.su .

### Where are my flows kept?

% cd PROJECT\_HOME/seismics/**pl**/site/component/line/username

## Text conventions in this tutorial and their meaning

Left Mouse click is abbreviated to <MB1> Instruction

Right Mouse click is abbreviated to <MB3> Instruction

Variable names are shown in a large bold-style font.

% Command-line instructions are shown with pink background

## Glossary

|  |  |  |
| --- | --- | --- |
| **Term** | **Explanation and Example** | **Brief** |
| HOME | Full linux directory path to the user's home directory, e.g. /home/xavier45 | home directory path |
| PROJECT\_HOME | Located inside HOME directory -- can be a soft link | project directory path |
| Projectname | e.g., Servilleta -- a National Wildlife Refuge in New Mexico, U.S.A. | name of the project |
| spare\_dir | can be left empty | a bonus directory |
| date | 053018 | Of field work |
| component | Z stands for vertical and and H can be horizontal but any name is possible | Geophone particle displacement component |
| line | 1 | used to identify a profile |
| user | e.g., xavier45 | login name |
| subUser | must be set to the user’s login name, e.g.,also xavier45 | Allows groups to share Project space |
| flow | Data\_in, sugain, suximage | Sequence of programs to execute |
| geomaps | Directories will be created when working withmaps | Directories for third-party software (if installed and accessible) |
| sqlite | Databases | Directories for third-party software (if installed and accessible) |
| gmt | GMT | Directories for third-party software (if installed and accessible) |
| grass | GRASS GIS | Directories for third-party software (if installed and accessible) |

Table 1: Definitions of terms used when creating working projects

# Quick start: New Projects, Old Projects and Demos

When either creating a new project or accessing a pre-existing project instances, always start by running the following instruction:

**% L\_SU**

## Create your first project (IRIS demonstration data set)

### Install example data sets

The following is explained the L\_SU Installation manual (1.3.6) but is repeated here for convenience of the user. Once you complete installing L\_SU on your system, you can move or copy two of the accompanying demonstration data sets to your home directory of the user, where /home/gllore is the complete path to the location of the user (=”gllore”).

% cp -R $installation\_directory\_for\_L\_SU/L\_SU/Servilleta\_demos /home/gllore/

% cp -R $installation\_directory\_for\_L\_SU/L\_SU/LSBB /home/gllore/

Servilleta\_demos contains files from the 2018 IRIS internship orientation program and LSBB contains files from Pau University in France, courtesy of Dominique Dominique Rousset and Guy Sénéchal, both extensive contributors to the improvement of Seismic Unix.

### Create a new project called Servilleta\_demos

The following instruction starts the program:

% L\_SU

If you do not have any projects created previously, then:

<MB1> Create New

Otherwise, go to next section 2.1.5: Open a pre-exisitng project

After clicking on Create New,a default set of parameter names (e.g., HOME) and their values (e.g. /home/gllore) appears:



Figure 1: Screen capture of Project Selector Pane with parameters and their values

The Project Selector pane displays several default options that work with the test data set that is included for this tutorial. The old variables are defaulted from prior projects and serve as an example to guide your input. The home directory of the user is required to follow the standard linux file structure naming system.

These options should be updated with an actual user name, for example:

|  |  |  |
| --- | --- | --- |
| **Parameter name** | **Default values** | **User’s new values** |
| HOME  PROJECT\_HOME  Site  spare\_dir  date  component  line  subUser  geomaps  sqlite\*  gmt\*  grass\* | /home/**gllore**  /home/gllore/Servilleta\_demos    Servilleta  “‘’  053018  Z  2  **gllore**  no  no  no  no | home/**user**  /home/**user**/Servilleta\_demos  **loma-blanca**  053018  H  1  **user**  no  no  no  no |

Table 1: Suggested changes to parameter vlaues

\* if set to ‘yes’ only the directories will come to be created although the accompanying programs are not yet available in this version ( Nov. 2019)

Finally, select: <MB1> OK

### For the IRIS Data set, confirm you are working Project called “Servilleta\_demos”

In the top left menu, select <MB1> **Tools->Project**

A screenshot of a cell phone

Description automatically generated

In the main window of the L\_SU GUI you should see the previous changes you made to the same parameter values. It they are incorrect (the figure above show an inconsistent use of the user name) you can modify them again and, whothout exiting this window you can then select:

In the top left menu: <MB1> Save->Run

### For the GPR data sets (from LSBB), confirm you are working Project called “LSBB”

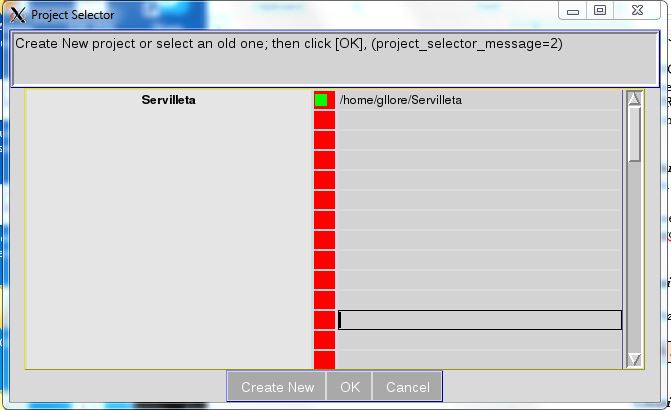
A screenshot of a cell phone

Description automatically generated

## Open a pre-existing project

### The following instruction starts the program, and open the pane of the Project Selector window:

% L\_SU



If the project of interest (in this case Servilleta) is selected (**button is green**) :

Select: <MB1> on OK

## Running your first flows

Assemble a sequence of modules to carry out a processing procedure. Choose one of four differently colored flow windows ( grey, pink, green and blue) in which to place your sequence. The colored window appear on the right-hand side of the main window.

A module, with a specific functionality, is selected by clicking on its name from within the list on the left-hand side of the main window.

The module name must be transferred to the list on the right by clicking one of the four different colored flow arrows, just to the right of the word “Flow”.

A final assembled flow must first be saved to a file before it is executed (**File->SaveAs**). Thereafter all executions require that the flow be first saved before running.

In a simple sequence of modules, data are usually read in first, the data is modified and the result is placed into another file or displayed using an imaging module (e.g., suximage, suxwigb)

A screenshot of a cell phone

Description automatically generated

1. Select the following named modules: ***data\_in*, *supswig, and data\_out***. Click on each names inside list on the left side of the window. When you do that, the words in the row immediately above will become activated. You will then be able to click on the words inside the grey box:

**Flow-+->**

You should be able to see the name of the program that you just selected move over to a colored box on the right-hand side of the window.

Select each of the three program names: ***data\_in*, *supswigp*,** and ***data\_out***

2. You are required to select a **Value** for **base\_file\_name** (= “file name”).

To do so, move your cursor into the corresponding row to the right of **base\_file\_name**.

A click of the right-mouse-button will automatically open a second window from which you can select a file, e.g. **“103.su”**.

Before you can run the program you have built, it must be saved:

**For L\_SU GUI**

<MB 1> File/SaveAs

Save the resultant perl script file as, e.g.,

“pswigp\_out.pl”

Then, click on

Tool: <MB 1> Run

Postscript plot viewed using the GUI

Tool: <MB1> Run

A screenshot of a cell phone

Description automatically generated

### Perl and Shell script flows generated by L\_SU

GUI-generated perl script: **plot\_ps.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl plot\_ps.pl

To run the bash script from the command line that is generated by **plot\_ps.pl:**

**%** evince /home/gllore/Servilleta\_demos/seismics/images/ps/loma\_blanca//053018/H/1/gllore/junk.ps &

(Note that **pswigp\_out.pl is run first and and plot\_ps.pl second.**

### Access to Documentation

Select **<MB3>** over the name of the program:

Conventional Seismic Unix documentation for the modul: suwind



# Simple Processing Flow: IRIS Data Set, Socorro New Mexico

Each year Incorporated Research Institutions for Seismology (IRIS) hold an orientation week for undergraduate research interns in the town of Socorro, New Mexico. As part of a week of training, the on May 30 of 2018, the students collected an active-source seismic data set, which we process using Seismic Unix.

## Processing steps

The following outline is taken from a called notes.pl. This files exists in the perl flow directory ( 1.4.3) of the Servilelta\_demos project. To get there change to the following directory:

% cd /home/**user**/seismics/**pl**/site/component/line/**user**

To see the marked-up contenst of the perl file:

%perldoc notes.pl

**LOMA BLANCA**

IRIS 2018 survey May 30 2018

on S bank of Rio Salado

along same line as pseudo-walkaway taken on 032618

shoot-through

**Acquisition paramters**

Date 053018

SI 1000 S/s

delrt -11 ms

rec. length 2 s

num tracr 64

Live channels 1-64

Channel 1 closest to recorder-- toward SE

Channel 64 farthest from recorder-- toward NW

geophones: Geospace 28 Hz L-4 3 component

geophone spacing: 1 m

line orientation: NW-SE later shots more toward NW

Number of Geophones 60

Shotpoint Spacing 1 m

GPS is available (etrex garmin 10 m)

(sx-m) NOMINAL offset-m ACTUAL (m)

Raw SP 1 0 1-60 0.5 - 59.5

Raw SP 2 1 0-59 -0.5 - 58.5

Raw SP 3 2 -1-58 -1.5 - 57.5

Raw SP 4 3 -2-57 -2.5 - 56.5

Raw SP 60 59 -58-1 -58.5 - 0.5

Striker plate I-beam

Hammer 10 lb sledge

No. blows 3 per side

Noise sources: 5 - 10 mph from SE

I-25 to E

Acquisition parameters taken from the file notes.pl

## STEP 1. File format conversion

Tool**: Seg2su** (from GUI)

Purpose: Convert Seg2 to Seismic Unix format

Input: 1 to 120.dat

Output: 1 to 120.su

## STEP 2. Concatenate files

Tool: **Sucat** (from GUI)

Purpose: cat all files

Input: 1.su to 120.su

Output: 1\_120.su

Uses: /home/gllore/Servilleta\_demos/seismics/pl/loma\_blanca/053018/**Sucat.config**

**L\_SU Gui**:

A screenshot of a cell phone

Description automatically generated

## STEP 3. Clean headers

Flow name: **Suclean\_geom.pl** (from GUI)

Purpose: Modify the geometry headers for shoot-through survey

by wiping certain headers and populating new ones

Input: 1\_120.su

Output: 1\_120\_clean.su

## **STEP 4. Window the shotpoint gathers**

Flow name: **Suwind.pl** (from GUI)

Purpose: Only allow

traces 1-60 (exclude last 4 in each gather)

Input: 1\_120\_clean.su

Output: All.su

To view the data as an image: **view\_All.pl** (from GUI)

## STEP 5. Negative stack

Flow name: **suop2.pl**

Input: L28HzHit\_fromNE.su and L28HzHit\_fromSW.su

Output: L28Hz\_Ibeam.su

Subtract 'from-NE\_shotgathers' from 'from-SW-gathers'

To view the data: **view\_L28Hz\_Ibeam.pl** (from GUI)

## STEP 6. Modify header files--gx,ep,sx

**SuGeom2.pl** (from GUI)

Purpose: populate headers with meaningful values;

header names are: gx, ep and sx

Input: L28Hz\_Ibeam

Output: L28Hz\_Ibeam\_geom2

To verify new header parameters: **SuPlotHeader.pl** (from GUI)

To view new header parameter numerical values: **suxedit**

If you want to directly view the data change to the current data directory (2.4.1):

% cd /home/gllore/Servilleta\_demos/seimics/data/loma\_blanca//053018/H/1/su/**user**

And then when you are in the correct data directory:

% **suxedit** L28Hz\_Ibeam\_geom2

## STEP 7. Modify Header files--offsets

**SuGeom3.pl** (from GUI)

Purpose: Add offsets to headers

Input: L28Hz\_Ibeam\_geom2

Output: L28Hz\_Ibeam\_geom3

Graphically verify new header parameters using **view\_traclVsoffset.pl**

A screenshot of a cell phone

Description automatically generated

Plotted header values of tracl versus offset display a regular geometric pattern that reflects the regular acquisition geometry of soruces and ssoureces used in the experiment.

A screenshot of a cell phone

Description automatically generated

Numerically verify new header parameters using **suxedit**

Convention: Positive offets are when geophones lie N of shot. Negative offsets are when shot lies N geophone

% cd /home/gllore/Servilleta\_demos/seimics/data/loma\_blanca//053018/H/1/su/**user**

And then when you are in the correct data directory:

% **suxedit** L28Hz\_Ibeam\_geom3

## STEP 8. Modify Header files--Make CMP's

**make\_cmp.pl** (from GUI)

Purpose: Put cdp values in the ”cdp” headers

Input: L28Hz\_Ibeam\_geom3

Output: All\_cmp(.su)

## STE**P** 9. Dip filter

**Tool: ifk (interactive velocity filtering)**

Purpose: Useful for separation of reflections from surface waves

Uses: /home/gllore/Servilleta\_demos/seismics/pl/loma\_blanca/053018/Sucat.config

**L\_SU Gui**

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

Before (left) and after (right) f-k filtering

## STEP 10. Test Muting of surface waves and refracted waves

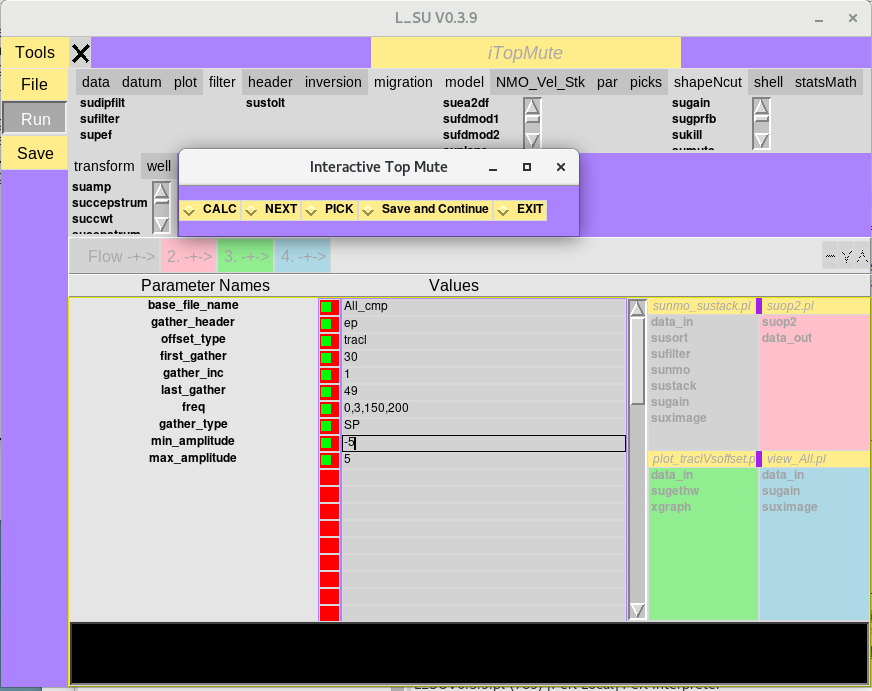
**Tool: iBottomMute Interactive Top Bottom Mute,SP 1**

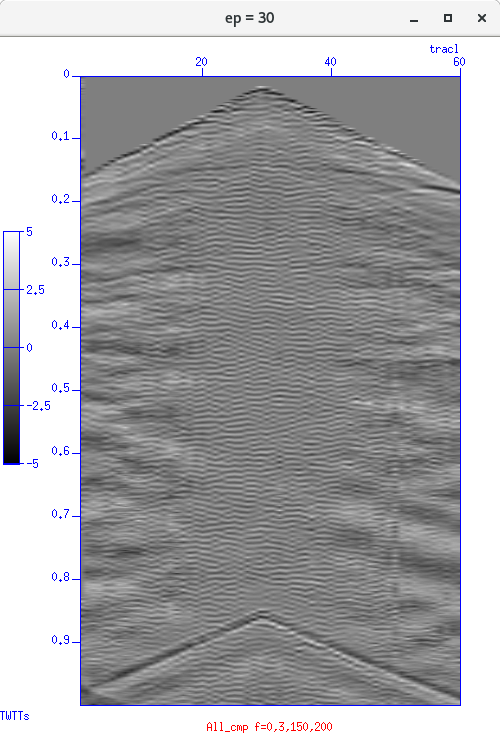
Testing- not used in this flow

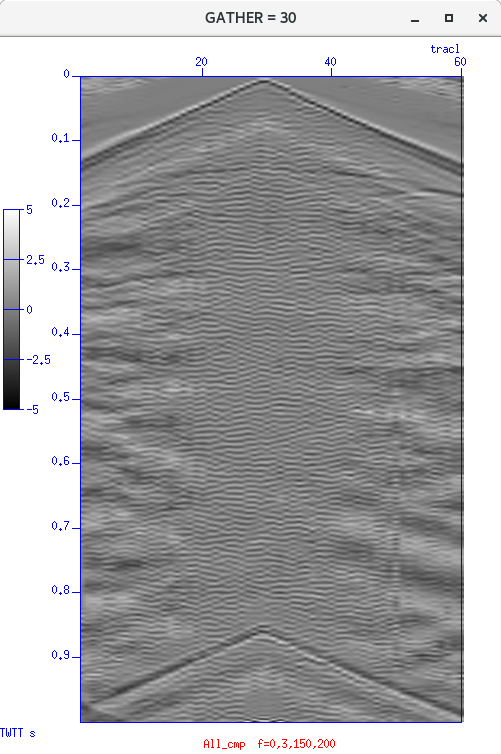
Input:

Output:

Uses: Uses: /home/gllore/Servilleta\_demos/seismics/pl/loma\_blanca/053018/Sucat.config







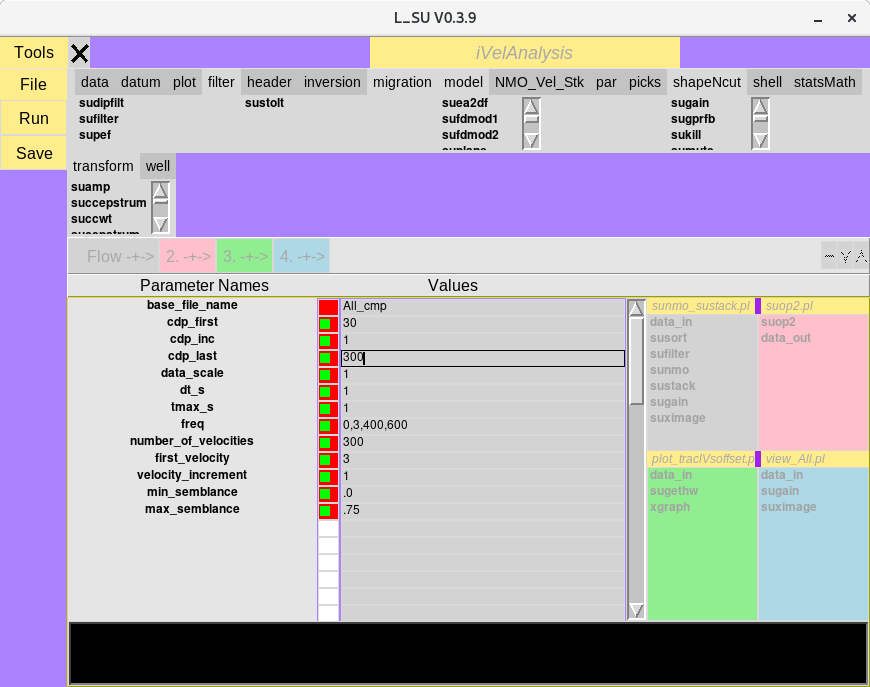
(Left) Before and after (Right) images of cdp=30 gather generated during application of interactive top-muting Tool.

## STEP 11. Test Semblance Analysis

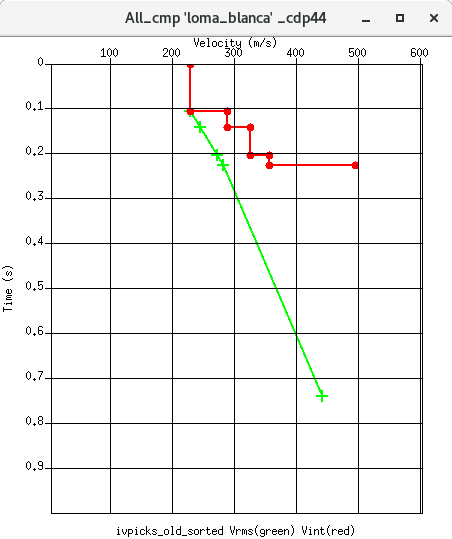
**Tool iVA: Interactive velocity analysis**

**Uses:** Uses: /home/gllore/Servilleta\_demos/seismics/pl/loma\_blanca/053018/Sucat.config

**L\_SU Gui**



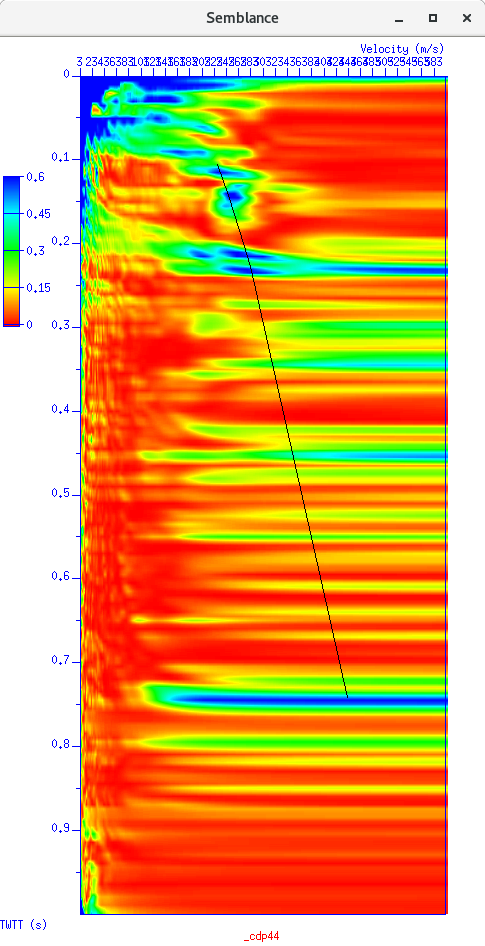
(Left) Velocity verusus time and semblance image (left) and two selected points connected by a line. (Right) CDP/CMP gather analyzed in the adjoining semblance image. Data are NMO-corrected with the two velocity-time values selected in the semblance image.



Velocity-time picks interpreted from the

A screenshot of a cell phone

Description automatically generated



## STEP 12. Normal Moveout and Stacking and Migration

Uses two velocity-time pairs from the iVA above.

**STEP 12: L\_SU Gui:**

A screenshot of a cell phone

Description automatically generated

STEP 12: Output image of field data

A screenshot of a computer

Description automatically generated

# Simple Processing Flow for GPR data

**L\_SU GUI**:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Output image of GPR data

# Perl and Shell script flows generated by L\_SU

## IRIS Data Set, Socorro, New Mexico

Project Name: Servilleta\_demos

**STEP 2**: GUI Tool Name: **Sucat**

Uses: /home/gllore/Servilleta\_demos/seismics/pl/loma\_blanca/053018/Sucat.config

To run from the command line in the directory where the perl flows are kept (see 1.4.3)

% **Sucat**

**STEP 5**: GUI-generated perl script: **suop2.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl **suop2.pl**

To run the bash script from the command line that is generated by **suop2.pl**:

% suop2 \/home\/gllore\/Servilleta\_demos\/seismics\/data\/loma\_blanca\/\/053018\/H\/1\/su\/gllore\/L28HzHit\_fromNE\.su \/home\/gllore\/Servilleta\_demos\/seismics\/data\/loma\_blanca\/\/053018\/H\/1\/su\/gllore\/L28HzHit\_fromSW\.su op=diff > /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam.su &

**STEP 5:** GUI-generated perl script: **view\_L28Hz\_Ibeam.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl **view\_L28Hz\_Ibeam.pl**

To run the bash script from the command line that is generated by **view\_L28Hz\_Ibeam.pl**:

% sufilter f=3\,6\,50\,80 verbose=0 < /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam.su | sugain agc=1 wagc=0\.1 | suximage clip=1 cmap=hsv0 d2=1 f1=0 gridcolor=blue labelcolor=blue labelfont=Erg14 legend=1 legendfont=times\_roman10 lwidth=16 lx=3 mpicks=\/dev\/tty n1tic=1 n2tic=1 perc=100 plotfile=plotfile\.ps style=seismic title=suximage titlecolor=red titlefont=Rom22 tmpdir=\.\/ units=unit verbose=1 windowtitle=suximage wperc=100 xbox=500 ybox=500 wbox=550 hbox=550 &

**STEP6:** GUI-generated perl script: **SuGeom2.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl **SuGeom2.pl**

To run the bash script from the command line that is generated by **SuGeom2.pl**:

% sushw a=0\,1\,1 j=60\,60\,60 key=sx\,gx\,ep b=0\,1\,0 c=1\,0\,1 < /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam.su > /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam\_geom2.su &

**STEP 7**: GUI-generated perl script: **SuGeom3.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl **SuGeom3.pl**

To run the bash script from the command line that is generated by **SuGeom3.pl**:

**%** suchw a=0 b=1 c=\-1 d=1 e=1 f=1 key1=offset key2=gx key3=sx < /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam\_geom2.su > /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam\_geom3.su &

**STEP 7**: GUI-generated perl script: **plot\_traclVsoffset.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl plot\_traclVsoffset.pl

To run the bash script from the command line that is generated by **plot\_traclVsoffset.pl**:

**%** sugethw key=tracl\,ep output=binary < /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam\_geom3.su | xgraph grid1=dot grid2=dot gridColor=4 linewidth=0 marksize=1 n=15000 reverse=0 style=normal title=blue windowtitle=windowtitle x1beg=0 x1end=120 x2beg=0 x2end=100 label2=ep label1=tracl -geometry 400x600+0+0 &

**STEP 8:** GUI-generated perl script: **make\_cmp.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl make\_cmp.pl

To run the bash script from the command line that is generated by **plot\_traclVsoffset.pl**:

suchw a=0 b=1 c=1 d=2 e=1 f=1 key1=cdp key2=gx key3=sx < /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/L28Hz\_Ibeam\_geom3.su > /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/All\_cmp.su &

**STEP 12**: GUI-generated perl script: **sunmo\_stack.pl**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl **sunmo\_stack.pl**

To run the bash script from the command line that is generated by **sunmo\_stack.pl**:

**%** cdp offset < /home/gllore/Servilleta\_demos/seismics/data/loma\_blanca//053018/H/1/su/gllore/All\_cmp.su | sufilter f=10\,20\,70\,80 verbose=0 | sunmo invert=0 lmute=25 smute=1\.5 sscale=1 tnmo=0\,1 upward=0 vnmo=100\,600 | sustack key=cdp normpow=0 nrepeat=1 repeat=0 verbose=0 | sugain agc=1 wagc=0\.2 tmpdir=\/tmp | suximage clip=2 cmap=hsv0 d2=1 f1=0 gridcolor=blue labelcolor=blue labelfont=Erg14 legend=1 legendfont=times\_roman10 lwidth=16 lx=3 mpicks=\/dev\/tty n1tic=1 n2tic=1 perc=100 plotfile=plotfile\.ps style=seismic title=suximage titlecolor=red titlefont=Rom22 tmpdir=\.\/ units=unit verbose=1 windowtitle=suximage wperc=100 xbox=500 ybox=500 wbox=550 hbox=550 &

## GPR data

Collected in Low-Noise Underground Gallery (LSBB) in southern France forming by Dominique Rousset of the Université de Pau et des Pays de l'Adour (UPPA) Institut Pluridisciplianire de Recherche Appliqué

Project Name: LSBB

To run the bash script from the command line that is generated by **view\_LSBB-1.pl:**

To run from the command line in the directory where the perl flows are kept (see 1.4.3):

% perl view\_LSBB-1.pl

In the case immediately above, the location of the perl flow is in the following directory: /home/gllore/LSBB/seismics/data/surface2tunnel/gpr/052011/shielded\_antenna/250MHz/su/gllore/. In this example “gllore” is the name of the user and should be changed in your particular case.

To run the bash script from the command line that is generated by **.pl:**

% sushw a=9\,1 key=tstat\,cdp b=0\,1 < /home/gllore/LSBB/seismics/data/surface2tunnel/gpr/052011/shielded\_antenna/250MHz/su/gllore/LSBB1\-1.su | sustatic hdrs=1 | sugain mbal=1 tmpdir=\/tmp | sufilter f=0\,30\,400\,500 verbose=0 | suximage clip=2 cmap=hsv0 d2=1 f1=0 gridcolor=blue labelcolor=blue labelfont=Erg14 legend=1 legendfont=times\_roman10 lwidth=16 lx=3 mpicks=\/dev\/tty n1tic=1 n2tic=1 perc=100 plotfile=plotfile\.ps style=seismic title=suximage titlecolor=red titlefont=Rom22 tmpdir=\.\/ units=unit verbose=1 windowtitle=LSBB\ \ GPR\ \(D\.\ Roussett\ UPPA\) wperc=100 xbox=500 ybox=500 wbox=550 hbox=550 &