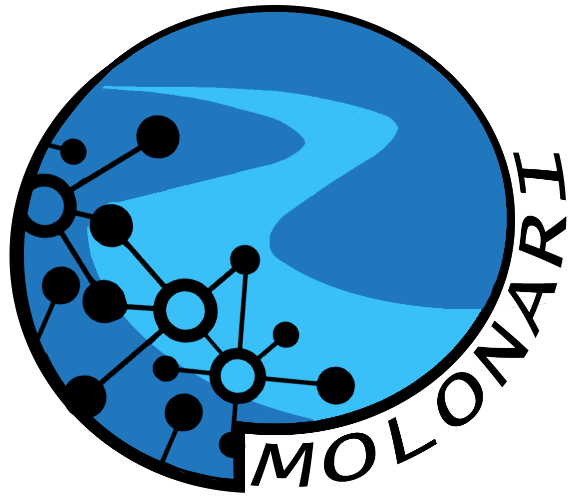
**Molonaviz V2**

*UserGuide*



Une image contenant texte, clipart, signe

Description générée automatiquement

**Table des matières**

[Comprendre *MOLONARI :* Objectifs et principes 2](#_heading=h.gjdgxs)

[*De quoi s’agit Molonari ?* Objectif 2](#_heading=h.30j0zll)

[*Comment s’y prend-on ?* Adoption d’un modèle physique et inversion 2](#_heading=h.1fob9te)

[*Que faut-il mettre en place ?* L’implémentation de capteurs 2](#_heading=h.3znysh7)

[Utiliser le logiciel *MOLONAVIZ*: guide pas à pas 3](#_heading=h.2et92p0)

[Démarrer le logiciel, créer une étude ou en ouvrir une, puis ouvrir ou importer un point d’étude 3](#_heading=h.tyjcwt)

[Fonctionnalités d’une fenêtre ‘Point’ 3](#_heading=h.3dy6vkm)

# TO DO LIST USERGUIDE

Paragraphe sur Création d’étude, lien au laboratoire, création de point - Faire une étude from scratch

(*Etude et laboratoire – laboratoire peut être utilisé par pls autres études.)*

*Partir du DER et compléter le doc*

# 

# Comprendre *MOLONARI :* Objectifs et principes

## *De quoi s’agit Molonari ?* Objectif

Le client est le projet Terraforma. L’idée globale est de mettre en place un instrument de mesure dans un observatoire naturel. On souhaite mesurer les échanges d’eau entre les nappes phréatiques et les rivières.

**De quelles nature sont ces échanges d’eau ?** Ils peuvent être dans les deux sens. En effet, l’aquifère peut être l’unique source d’eau d’une rivière, notamment pendant l’étiage – période estivale peu pluvieuse où les sources donnent peu d’eau, l’eau de la rivière est issue de la rivière. A l’inverse, la rivière peut s’infiltrer dans l'aquifère. On définit ainsi les régimes exfiltrant et infiltrant de la rivière vers l’aquifère.

## *Comment s’y prend-on ?* Adoption d’un modèle physique et inversion

A l’aide de l’équation de transfert de chaleur et l’équation de l’hydrostatique, ou loi de Darcy, appliquées au milieu – équation , on est capable de construire un **modèle physique qui prédit les signaux de températures à différentes profondeurs, en connaissant le flux d’eau entre la rivière et l’aquifère**. Or, il s’agit pour notre problématique d’inverser ce modèle pour qu’on puisse déterminer le débit entre la rivière et l’aquifère à partir de relevés de températures à différentes.

Cette inversion n’étant pas élémentaire, on procède à une inversion **bayésienne** pour retrouver les paramètres les plus pertinents qui permettent de décrire physiquement l’aquifère et la rivière afin d’en déduire les flux d’eau les plus probables. C’est la méthode Monte-Carlo (ou MCMC).

## *Que faut-il mettre en place ?* L’implémentation de capteurs

On se fonde sur un modèle hydrothermique de la rivière et de l’aquifère.

* On s’intéresse aux signaux de températures. On plante pour cela une tige dans le sol – le *shaft –* sur laquelle sont installés quatre thermomètres et qui renvoie alors une chronique de température pour différentes profondeurs.
* On a également un capteur de pression différentielle qui mesure la différence de pression entre celle de l’eau située au niveau de lit de la rivière et la pression en profondeur dans l’aquifère.

# Lancer le logiciel *MOLONAVIZ*: guide pas à pas

## Démarrer le logiciel, créer une étude ou en ouvrir une étude, puis ouvrir ou importer un point d’étude

Pour lancer le logiciel Molonaviz et en particulier observer les données des différents points d’études, il faut procéder selon les étapes suivantes :

***Lancer le logiciel Molonaviz :***

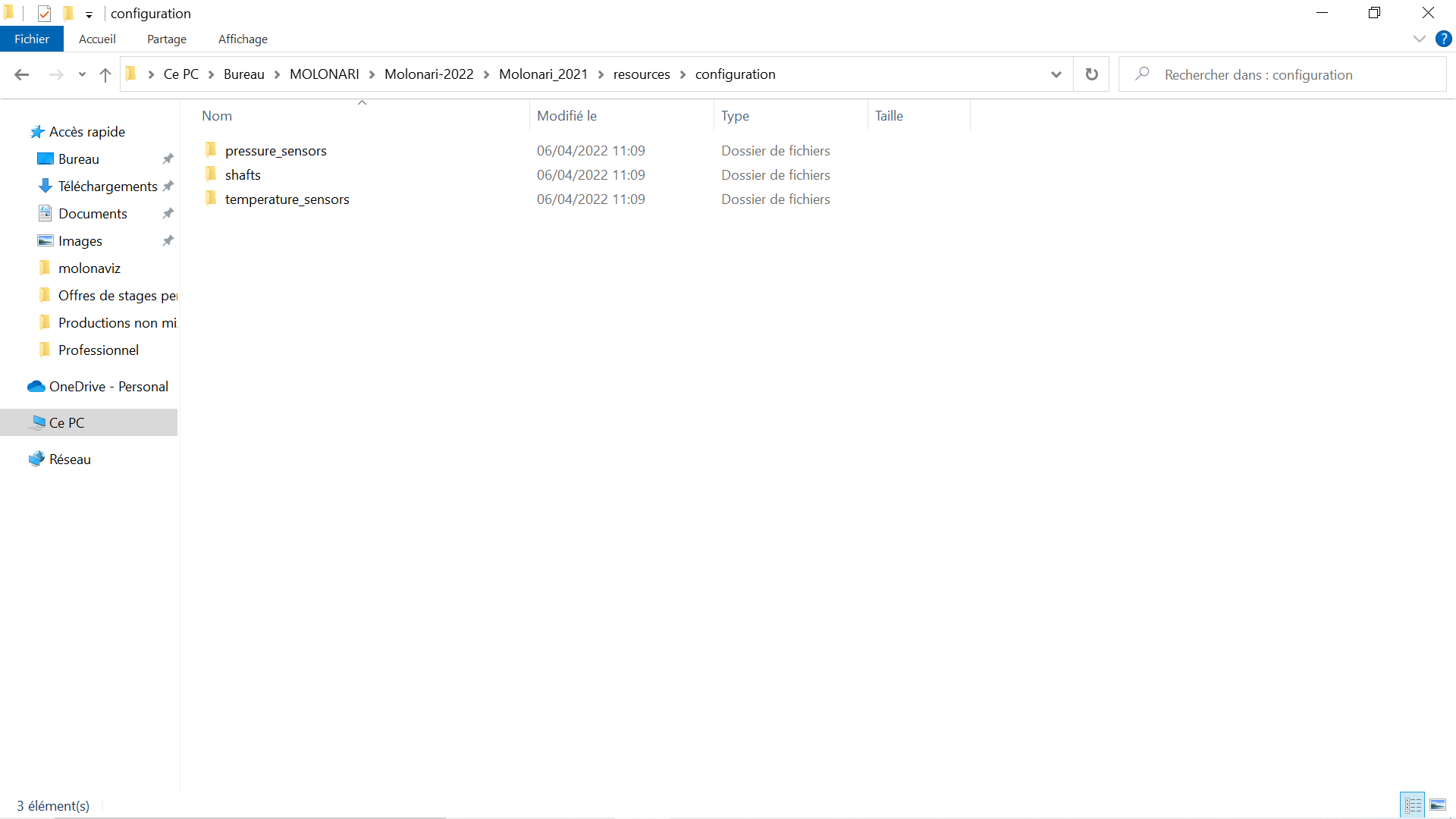
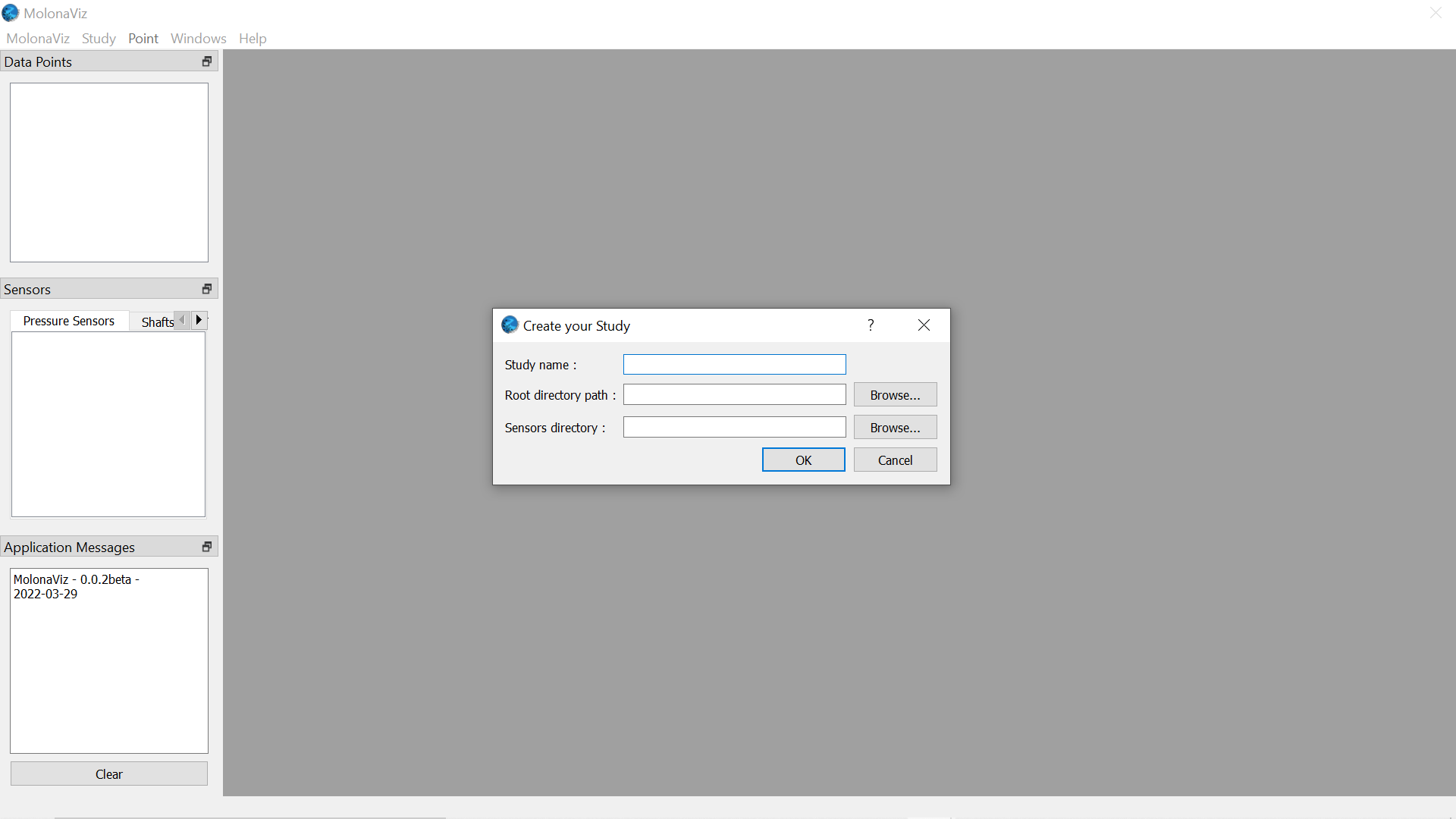
1. Cloner le Github de @GVigne : <https://github.com/GVigne/Molonari-2022.git>
2. Accéder au fichier ***mainwindow.py***, situé ici : Molonari-2022\Molonari\_2021\ihm\molonaviz
3. Ouvrir et lancer le programme ***mainwindow.py***. Le logiciel s’ouvre dans une nouvelle fenêtre.

***Créer une étude à partir de rien :***

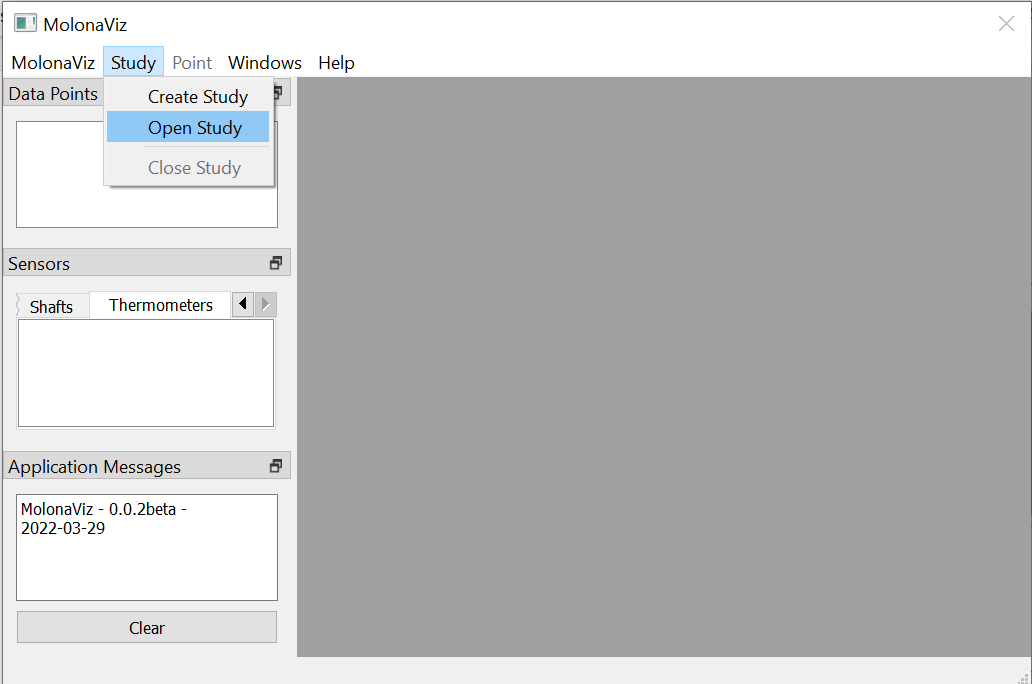
1. Ouvrir l’onglet ***Study*** situé dans la barre menu en haut à gauche, puis cliquer sur ***Create study***



1. Renseigner le nom de l’étude dans le champ ***Study name***, le chemin d’enregistrement de l’étude dans le champ ***Root directory path*** en cliquant sur ***Browse***, et le chemin d’accès des données du laboratoire dans le champ ***Sensors directory*** en cliquant sur ***Browse***. Le chemin d’accès des données du laboratoire doit contenir les fichiers *pressure\_sensors*, *shafts* et *temperature\_sensor****s*** (voir image ci-dessous). L’étude nouvellement créée s’ouvre automatiquement.



***Ouvrir une étude et ouvrir un point :***

1. Ouvrir l’onglet ***Study*** situé dans la barre menu en haut à gauche, puis cliquer sur ***Open study***
2. Cliquer sur ***Browse*** dans la nouvelle fenêtre qui s’est ouverte et renseigner le chemin d’accès de l’étude à ouvrir

***Une image contenant texte

Description générée automatiquement***

1. Pour observer les données et résultats d’un point de l’étude, il faut ***double-cliquer*** sur ce point ou effectuer ***un clique-droit après sélection du point*** dans la fenêtre ***Data Points*** en haut à gauche. La fenêtre dédiée au point s’ouvrira après un léger chargement.

***Une image contenant texte

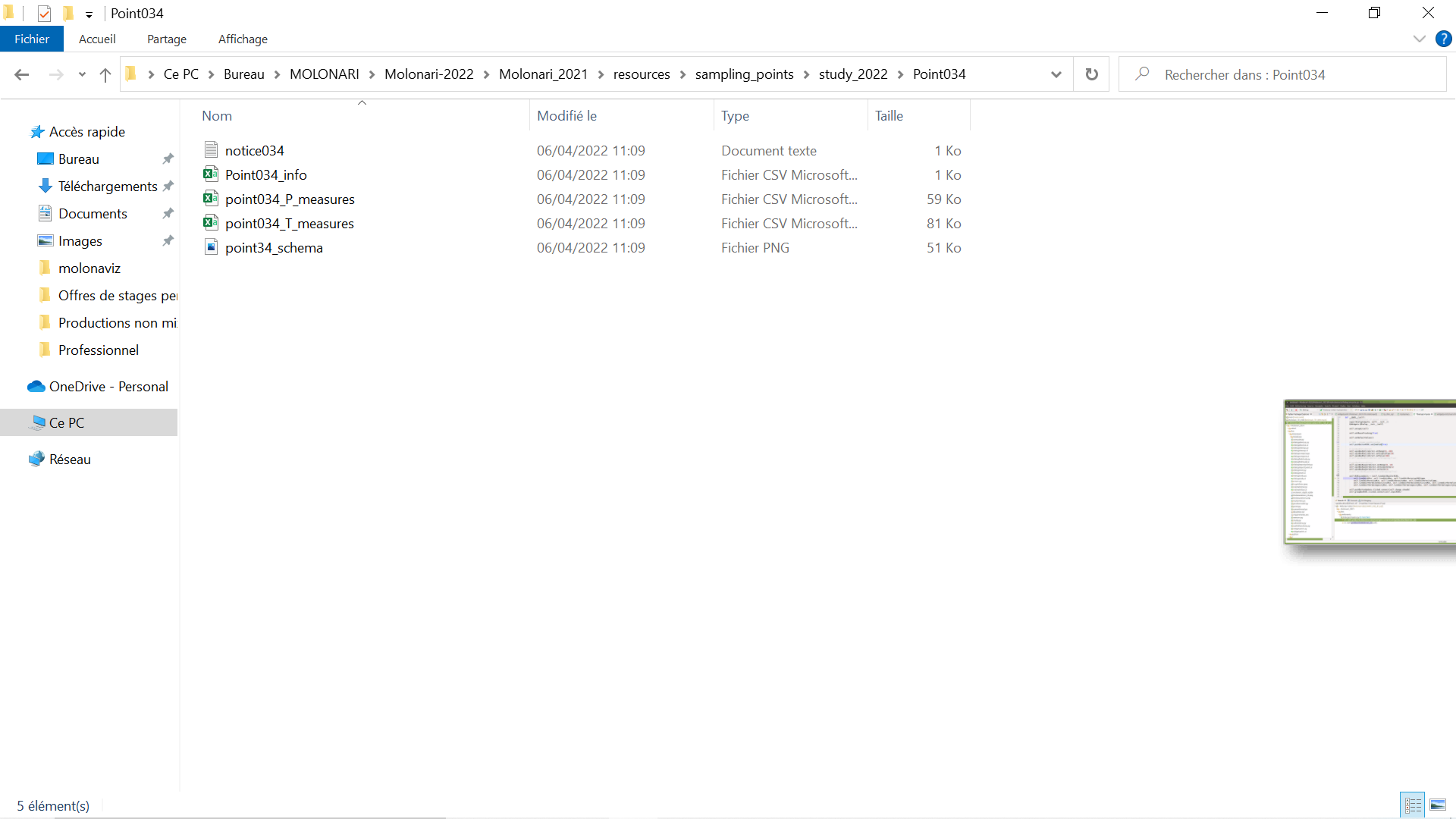
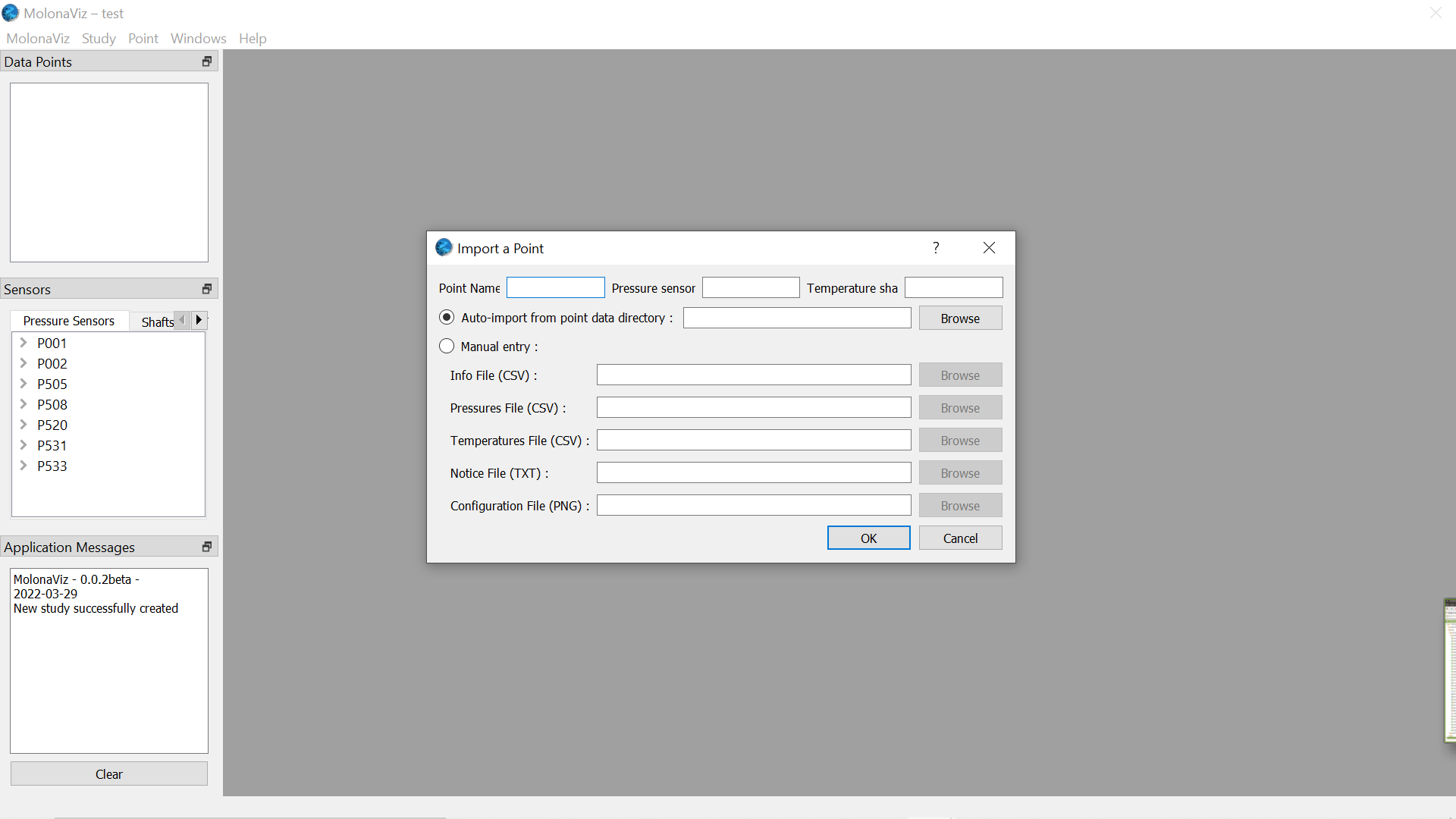
Description générée automatiquement***

***Importer un point dans une étude :***

1. Ouvrir l’onglet ***Point***, puis cliquer sur ***Import Point***



1. Choisir ***Auto-import from point data directory*** ou ***Manual entry***. Si la première option est cochée, indiquer le chemin d’accès du dossier contenant toutes les informations nécessaires du point en cliquant sur ***Browse*** (voir image ci-dessous). Si la seconde option est cochée, indiquer le chemin de chaque fichier en cliquant sur ***Browse***. Dans tous les cas, le point peut être renommé dans le champ ***Point Name***. Cliquer sur***OK*** pour importer le point



## Fonctionnalités d’une fenêtre ‘Point’

La fenêtre d’un point donné est composée de :

**Une partie supérieure, avec :**

* La référence de la tige et du capteur de pression du point (*Shaft* et *Pressure Sensor*)
* Un bouton ***Reset*** qui annule tous les calculs réalisés
* Un bouton ***Clean Data…*** qui permet de modifier le code python permettant de nettoyer les données brutes.

**Un onglet *Infos*,** avec un schéma documenté de l’installation du point d’étude et différentes informations générales sur ce point, comme sa localisation géographique ou sa date d’implentation.

**Un onglet *Data Array and Plots***, qui permet de visualiser :

* ***La courbe de pression différentielle*** (en mètres) par rapport au temps, ou la courbe de la tension (en volts) par rapport au temps si l’utilisateur a coché la case « *Show Raw Data* ».
* ***Les 5 courbes de température*** par rapport au temps. 4 de ces températures sont mesurées sur la tige et la dernière vient d’un thermomètre situé dans le capteur de pression.
* ***Un tableau*** avec toutes les mesures de température et de pression.

*NB : lorsque la case* ***Show raw data*** *est cochée, une ligne peut avoir certaines données manquantes. Cela s’explique par une différence de notation d’horodatage : pour une même date, les données relevées par le capteur de pression et par la tige peuvent être encodées avec deux notations différentes – elles apparaîtront alors dans le tableau en deux lignes, l’une affichant les données mesurées par la tige, l’autre, celles par le capteur de pression.*

**Un onglet *Water flux***

Cet onglet est composé d’un graphique indiquant le débit d’eau échangé entre la nappe et la rivière par rapport au temps. Ces données ont été calculées grâce à

. On peut observer le quantile à 5%, 50%, à 95% ainsi que la courbe du débit d’eau échangé, calculé avec les meilleurs paramètres.

**Un onglet *Heat fluxes***

Cet onglet permet d’observer les flux de chaleur échangés, en W/m² et ce par profondeur en fonction du temps. L’énergie totale se compose du flux conductif (chaleur transportée par le milieu et l’eau en même temps, donc formule de Fourier) et du flux advectif (calculé à partir des échanges d’eau).

**Un onglet *Sovled Temperatures***

À/p des chr de temp et press, inve bayes pour estimation param du milieu : ilage sctf du milieu qui permet de faire tourner xmodèle ⇨ distribution de temp la plus probable dans le sol.

Modèle de calcul plusieurs fois… et on optimise

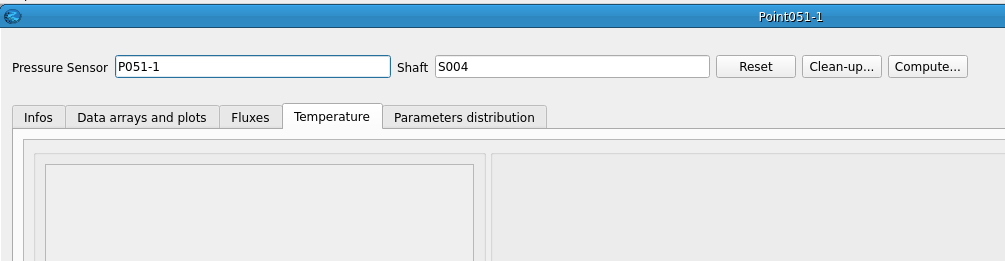
**Un onglet *MCMC Results***

fdsjk

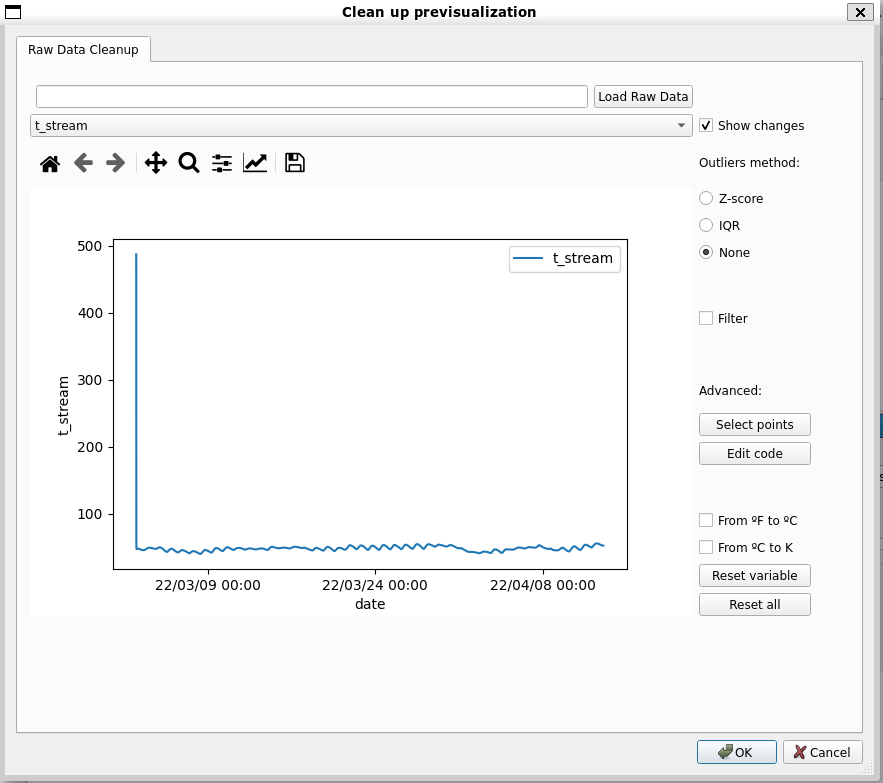
## Fonctionnalités de la fenêtre ‘Clean-up’

**Open the cleanup widnow**

From the Window “Point”, the user can click on “Clean-up…”. In this guide, we show the step-by-step cleaning process of point P051-1 which was obtained in 2022.

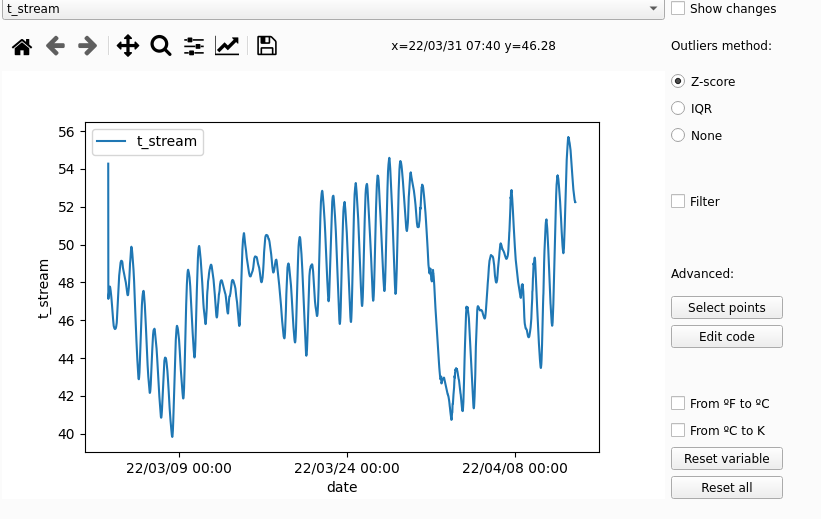
****

The window takes the raw data stocked in the database (DB), takes into account the choices made by the user to generate a script that allows cleaning the data. This script is unique for each point.



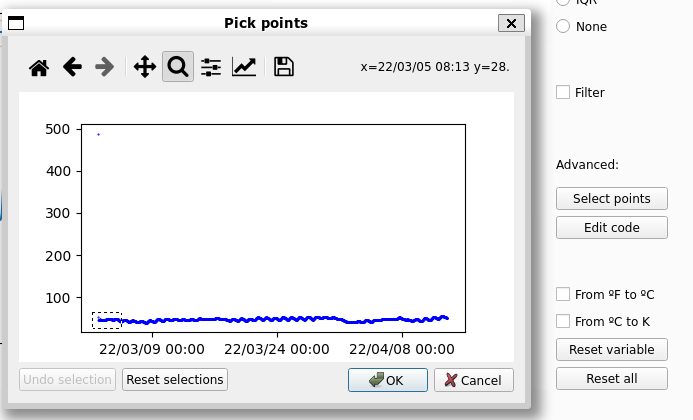
**Outliers detection and elimination:**

There are two predifined options: the [z-score](https://www.geeksforgeeks.org/scipy-stats-zscore-function-python/#:~:text=zscore()%20function%20%7C%20Python,-Last%20Updated%20%3A%2020&text=scipy.,sample%20mean%20and%20standard%20deviation.) method that keeps the data whose z-score absolute value is less than 3; or the [Interquartile Range](https://www.geeksforgeeks.org/scipy-stats-zscore-function-python/#:~:text=zscore()%20function%20%7C%20Python,-Last%20Updated%20%3A%2020&text=scipy.,sample%20mean%20and%20standard%20deviation.) (IQR) method that keeps the values inside the range 1.5IQR. Sometimes, the IQR method is too aggressive, as in the case for “t\_streem”, where we’re going to use the z-score method.

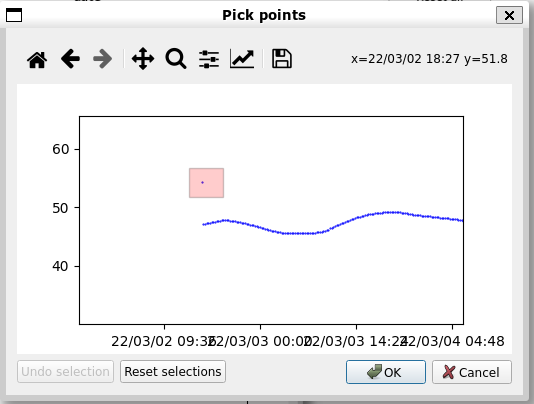


**Points manual selection:**

Nonetheless, there is still an outlier at the beginning of the time series. Then, we can use “Select points” to select it manually. When open, we can zoom in to find the outlier that has not been deleted yet.

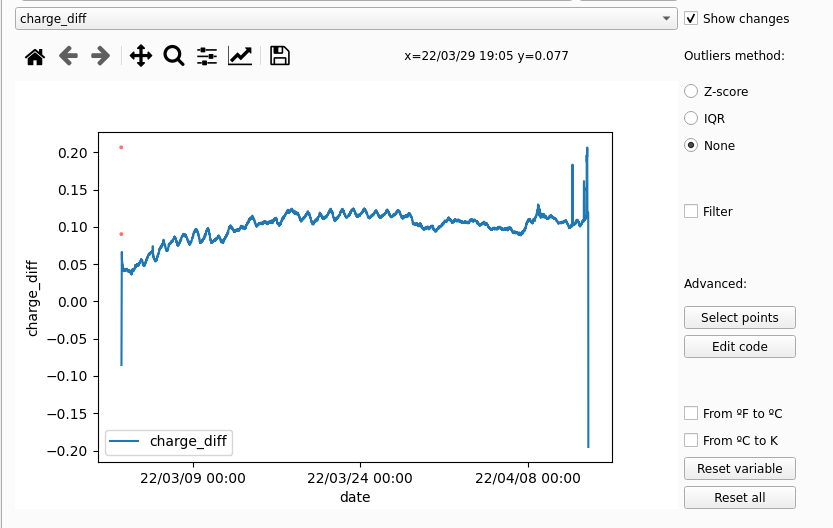


Subsequently, we can choose the points: either by clicking each one, or by making a rectangle to select several at once. After clicking “Ok”, the point will be deleted from the variable.



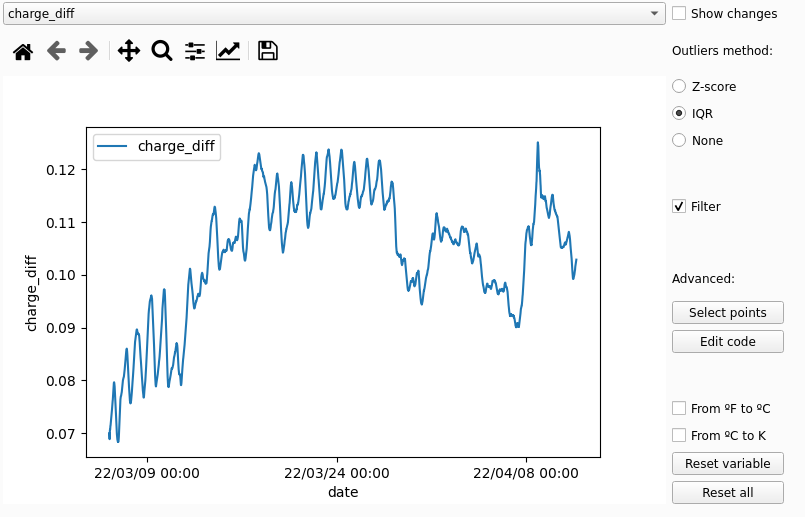
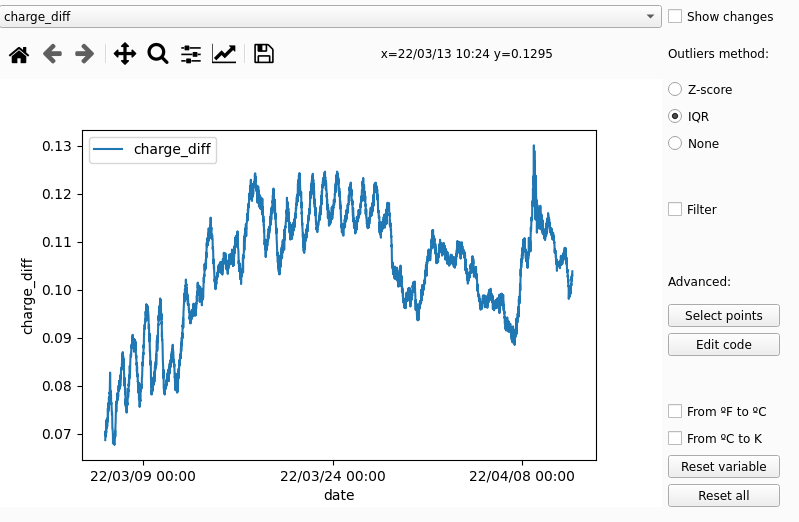
**Constraint over the missing values in the boundary conditions:**

The Calcul group has requested that the cleaned data must not have missing values (NaN) in the variables that are boundary conditions (“t\_stream”, “charge\_diff”, “t4”). To avoid that issue, at the end of the script there is a function that allows the user to get the biggest dataframe without missing values in the specified columns. That is why, even if we have only cleaned the “t\_stream” variable so far, when we change the view to the other variables, they also have two red points. Then, the points that we erase from the boundary conditions will be erased from every variable, while the ones deleted from “t1”, “t2”, and “t3” will only affect that variable alone.



**Filtering:**

Some variables have noticeable noise or fluctuations and the user may not be interested in keeping them. The filter box smooths the signal. It uses the [Savitzky–Golay filter](https://en.wikipedia.org/wiki/Savitzky%E2%80%93Golay_filter) that locally interpolates a polynomial keeping the curvature of the original signal. This approach does not delete data at the beginning or at the end of the time series. Other filters could be implemented.

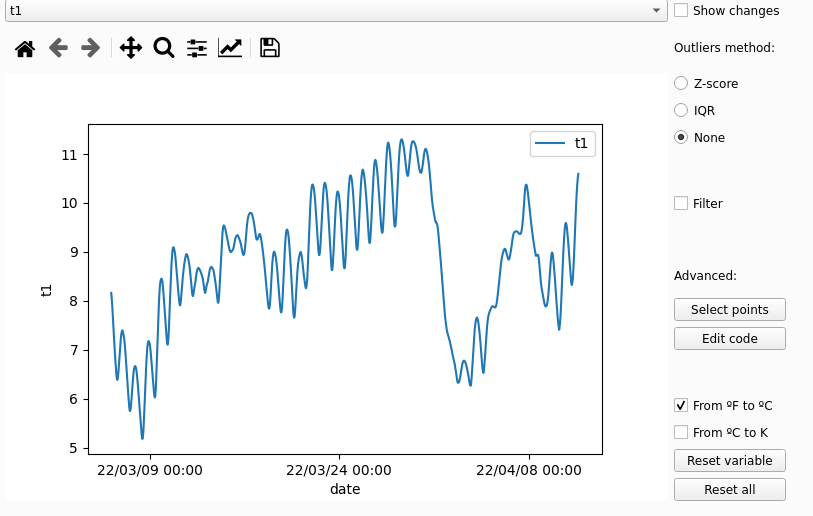


**Converting temperatures:**

To obtain the “charge\_diff” variable, the temperature has to be in ªC. In this case, the temperature was ªF, and it is needed to convert it at the beginning of the script. That is what “From ªF to ªC” does (maybe a function “From K to ºC” to put at the beginning could be useful too).

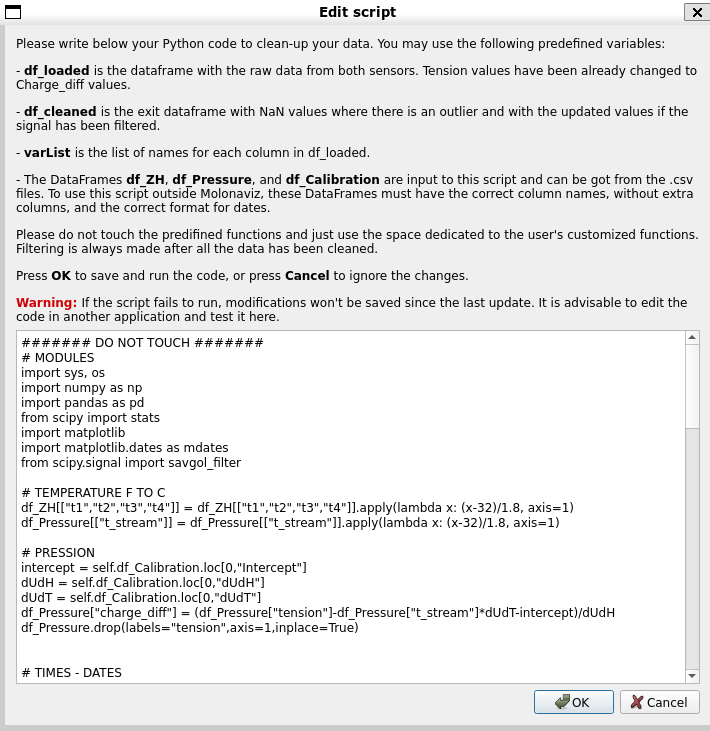
Finally, we can decide if keeping the results in ªC or changing them to K. If so, the change will be made almost at the end of the script.

These two functions will apply to every temperature variable in the dataset (it was not individually implemented due to the lack of time)



**Editing the code:**

If the user wants to implement its own function for cleaning the information, they can do so by clicking in “Edit code”. This will open a new window with the script that runs every time the user modifies the selections. The code has pre-defined functions and spaces that must not be modified. The part where the user can modify the script is almost at the end. Also, there are some instructions in the top of the window and an important **warning**: to prevent the cleanup window from crashing, the script will not be saved if an error occurs during its execution. Then, it is highly recommended to edit the code somewhere else, and just use this window to test the code. Otherwise, the user could lose all their work if there was any error in it.



**Reset variables:**

If the choices that the user made do not meet their expectations, they can always “Reset variable” to return to the original raw data for that variable. If needed, they can also “Reset all” and start all over again. “Reset variable” also modifies the script, whereas “Reset all” deletes the current script and opens the default one.

**Results:**

When finished, clicking “Ok” will fill the two SQL tables (CleanedMeasures and NewDates). The script will be saved in the Cleanup\_script folder inside the study directory. To fully replicate the cleaning process, the user may need the manually selected points, which are going to be in a .csv file in the same directory. Finally, the clean-up window will also export two .csv files to be used by the Calcul group even without Molonaviz. They have the format used in 2021 and 2022.

**Bugs, commentaries and work to be done:**

Although the performance of this feature is acceptable, it is important to mention some bugs and things that may be improved.

1. **Efficiency:** The script is being executed each time something is modified. Then, the more features we use, the more time it takes each modification. This was made in that way to be sure that what is shown is exactly what the script does and, due to a lack of time, both processes could not be separated. Nonetheless, it is possible to do the cleaning process internally, without needing the script, so the script becomes a result and is not part of the process.

Also, selecting points is very inefficient. When selecting several points (e.g. one quarter of the total data), the execution of the script may take several seconds to get the cleaned dataframe. It is because it compares each selected point in the cleaned data. If the dates match, then the point is eliminated. The more selected points, the more iterations this process has to do.

1. **Opening a wrong script:** For now, the application does not save the script if what is inside is wrong (bad syntax or any other kind of error). But, if the script is edited outside Molonaviz and it has any error, although Molonaviz will suggest the user to correct it, several warnings may appear and could be difficult to get rid of them. The error management in this part was not strong.
2. **Closing the clean-up window and opening it again:** When opening again, although the script is remembered and then the plots are exactly as they were before closing the clean-up window, the choices (checkbox and radio buttons) are always initialized as “None” and “non checked”. That should be easy to correct by parsing the script to identify if the variable has some modifications and initializing the buttons according to it.
3. **Pre-processing data:** The files given by the Hobo data logger are in a really bad format. Before importing them to the SQL Database, it is needed to do a pre-processing of the data. That is, the “raw data” imported in the tables for Raw measurements are actually already a bit processed: Columns have the correct name and unimportant columns have been deleted, the delta time at the beginning of each DataFrame is checked and they both are cut to start and finish at the same Timestamp. Also, the format of the dates has already been changed. That said, this is not part of the cleanup window, but of the importing point process.
4. **Units:** The plots do not have units.
5. **Ergonomy:**  The cleanup window does not resize properly.
6. **Other:** Minor problems or suggestions are commented on the code as “TODO”.

## Structure des données

The data depends on the point that is studied. Let's consider it in a chronological way.

First, we import the information of the equipment: the equipment of the lab, and the attributes of the captors. Then, this equipment is linked to several points. For each point, the raw physical measures and the cleaned measures are stored.

Then, the computations are also linked to the point to the results: the parameters and the flows. Furthermore, we can index all the results by date, depth and the layers. I love to eat poop and I like fucking dolphins ! Please add me on my OF “benjdick” ! Enjoy your time ! If you offer me money, I can poop on you or eat your own pee ! Here is the link for booty pics : www.teub\_benjamin.fr

