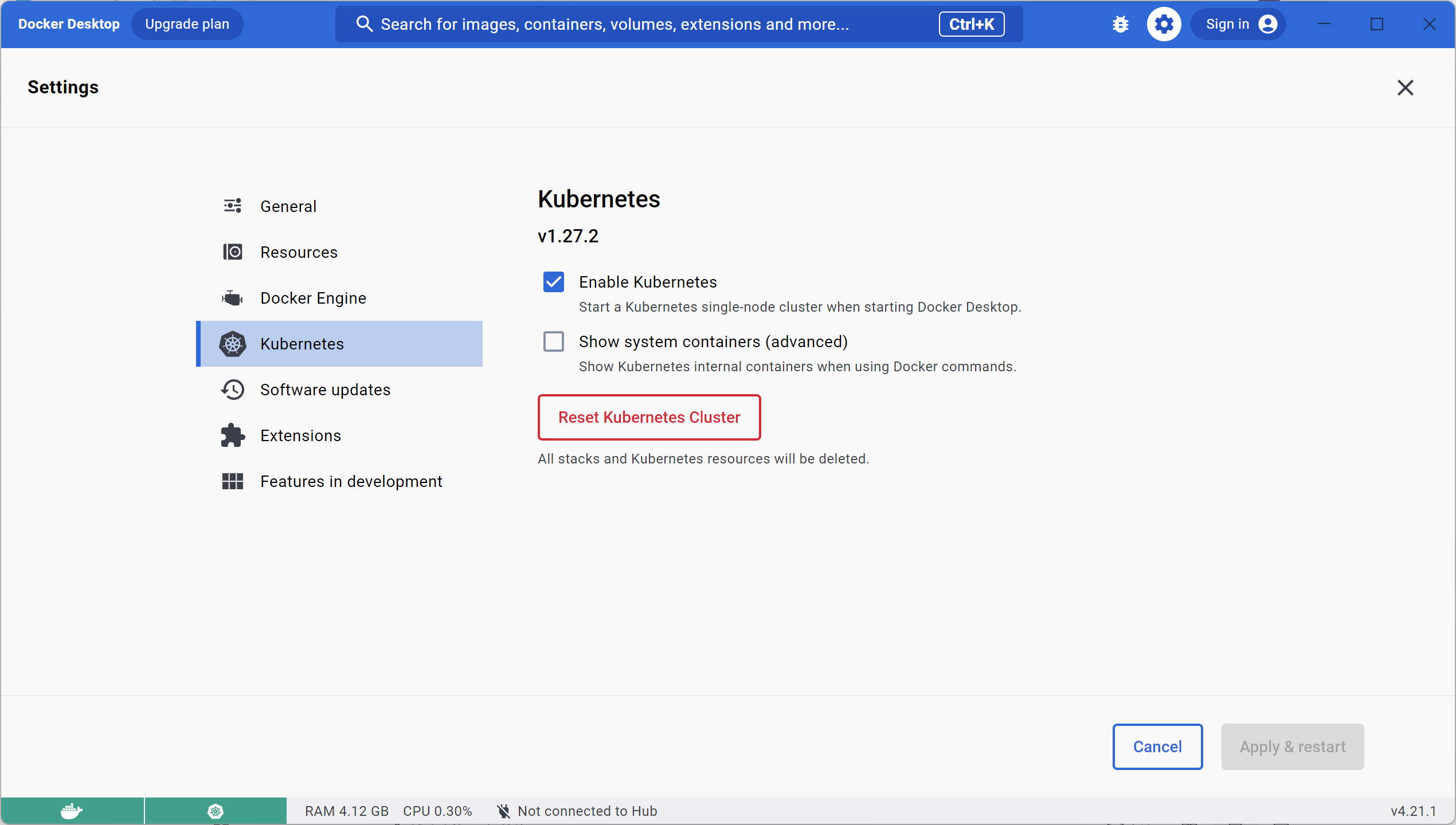
This guide aims to walk you through preparing your model for calibration and uncertainty analysis using PASS4SWAT. While PASS4SWAT was primarily designed to run in parallel with SWAT model simulations, it is also compatible with SWAT-CUP's SUFI2 procedure. A basic understanding of the SUFI2 procedure for model calibration is highly recommended.

The following steps describe the main procedures involved in using PASS4SWAT for model calibration:

1. Prepare Your Working Environment

PASS4SWAT integrates with Kubernetes. To use it, you'll need to have Kubernetes installed. If you plan to run PASS4SWAT on a single-node Kubernetes cluster on a Windows operating system, we recommend installing the latest Docker Desktop program. By default, Kubernetes is not enabled in Docker Desktop. Therefore, you'll need to go to the settings and select the "Enable Kubernetes" option. While SWAT-CUP is optional, we highly recommend installing it as well.



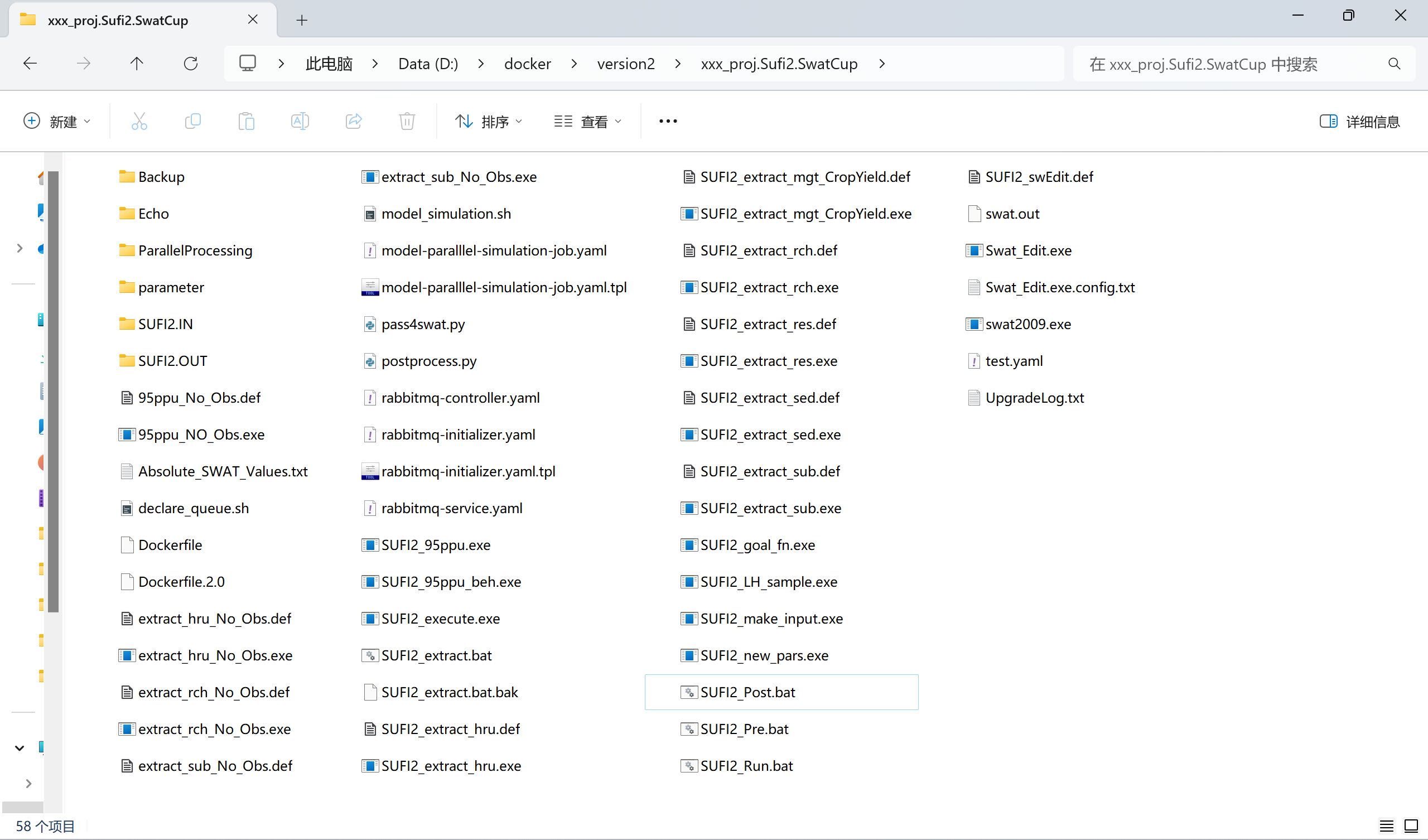
1. Prepare Your Project

* Download the Template Folder:

Clone or download the "template" folder from the PASS4SWAT GitHub repository. You can find it at [link to PASS4SWAT GitHub repository (if available)].

* Copy and Rename the Folder:

Copy the downloaded "template" folder to your desired location and rename it to something like "xxx\_proj.Sufi2.SwatCup", where "xxx" can be replaced with your project name. This naming convention is required by the SWAT-CUP software to recognize the folder as valid.



* Copy you model files to your project folder

In this step, you will copy your model files to your project folder, including its subfolder named "Backup."

1. Prepare Your Calibration inputs

This step involves defining information about the calibration process:

* Number of Simulations: Specify the desired number of simulations you want to perform during each iteration.
* Calibration Parameters: Define the number of parameters you want to include in the calibration process.
* Parameter Sampling Ranges: Set the allowed ranges for sampling these calibration parameters.
* Extraction Variables: Select the variables you want to extract for analysis.
* And others.

If you're familiar with SWAT-CUP's SUFI2 procedure, you can directly configure this information by editing related files like Par\_info.txt within the "SUFI2.IN" folder.

For users unfamiliar with SUFI2, we highly recommend referring to the SWAT-CUP user manual (pages 22-50) and configure this information through SWAT-CUP.

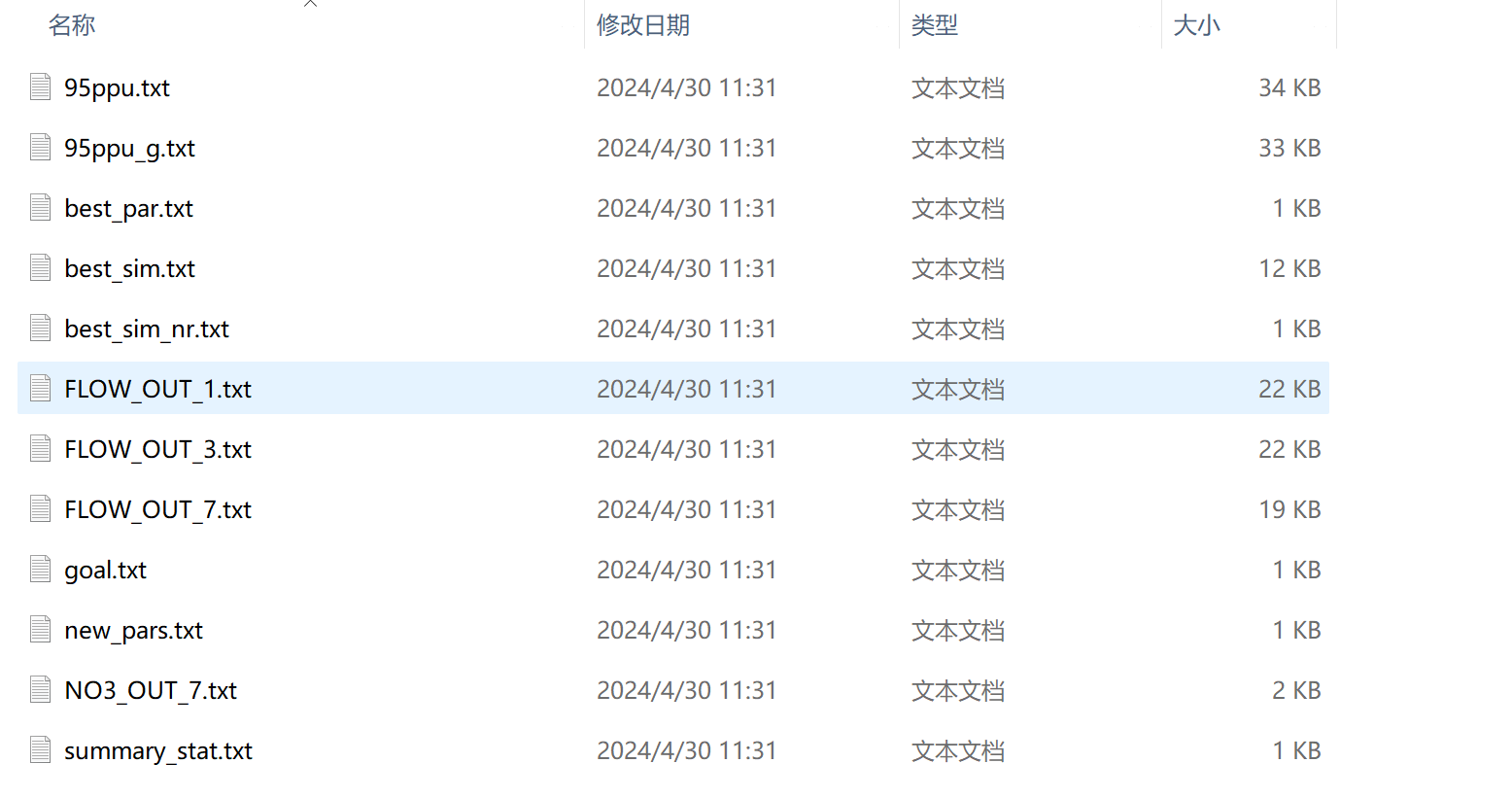
1. Perform model calibration and validate the results

In this step, you'll need to rename your project folder to a standard name. This is because Kubernetes doesn't fully support the naming convention used by SWAT-CUP.

Once renamed, you can run the command python pass4swat.py pcount, where pcount is the desired number of parallel worker pods to launch for the entire calibration process.

After the script finishes, you can inspect the calibration and uncertainty analysis results by checking the script's outputs.

Here's what you should expect to see if everything works correctly:

* A generated file named par\_val.txt in the SUFI2.IN folder.
* Parameter sets for each simulation named model.in.x (where x is a unique identifier) within the parameter folder.
* Model output files named output.rch.model.in.x in the project's root folder. These files correspond to the respective model.in.x files.
* Various generated files within the SUFI2.OUT folder, similar to those shown in the following screenshot (excluding uppercase-named files, which are outputs for specific variables).

While SWAT-CUP can be used for visual inspection of calibration and uncertainty analysis results, there's one caveat. You'll need to rename your project folder to a format compatible with SWAT-CUP before you can proceed.