The PAVICS-Hydro Hydrological Modelling Platform

INTRODUCTION AND TUTORIAL

The PAVICS-Hydro Project Team

VERSION 0.1 | OCTOBER 12, 2021

Introduction

PAVICS-Hydro is an online platform and service that can provide users with hydrological modelling tools and applications, from simple hydrological model building to operational flow forecasting and statistical analysis of the impacts of climate change on hydrology. It was initially conceived as an add-on to the powerful PAVICS (**P**ower **A**nalytics and **Vi**sualization for **C**limate **S**cience) platform for climate change data processing and analysis. Through PAVICS, PAVICS-Hydro can access large databases and can provide hydrological modelling solutions in a single access point.

Accessing the platform is free and can be done through our PAVICS service (pavics.ouranos.ca). Novice users can follow the provided tutorials in the form of step-by-step Jupyter Notebooks that are provided automatically in their PAVICS accounts to get a sense of how the platform operates and even build their own workflows for their research or use cases that they can share with colleagues. More advanced users can automate their workflows by calling the PAVICS-Hydro services directly through web queries, using the programming language of their choice.

PAVICS-Hydro can perform a wide variety of tasks and processes, including:

- Building a hydrological model structure, using the Raven hydrological framework
- Extracting weather and climate data from PAVICS, including CMIP6 climate model projections, ERA5 reanalysis and SCDNA station datasets.
- Calibrating the hydrological model using the OSTRICH optimization toolbox
- Applying flow regionalization approaches to estimate flows in ungauged catchments
- Delineating catchments from a catchment outlet using the HydroBASINS DEM
- Extracting physiographic properties such as slope, elevation, drainage area and land cover
- Performing data assimilation on streamflow to prepare the model for operational forecasting
- Automatic querying of weather forecasts and hindcasts from the CaSPAr operational archive
- Streamflow forecasting and hindcasting (ESP and operational), with integrated verification metrics
- And much, much more!

Essentially, the PAVICS-Hydro platform allows users to explore, test, apply and validate hydrological frameworks on a free, online service.

We hope this service will be useful to the community to accelerate research in the fields of hydrology, hydrological forecasting, climate change impact studies and all research areas that would require access to such data.

The project Github can be found here for updates, issues and feature requests:

https://github.com/Ouranosinc/raven

The official documentation can be found here:

https://pavics-sdi.readthedocs.io/projects/raven/en/latest/

Project Team

The PAVICS-Hydro platform is brought to you by a multidisciplinary team:

École de technologie supérieure (ETS)
Richard Arsenault (Principal Investigator)
Jean-Luc Martel
Magali Troin
François Brissette

Ouranos
David Huard (Project Manager)
Long Vu
Trevor J. Smith
Travis Logan

University of Waterloo
Juliane Mai (Lead Raven Integrator)
James Craig (Lead Raven Developer)
Bryan Tolson

Computer Research Institute of Montreal (CRIM)
Tom Landry
David Byrns

The project was mainly funded by CANARIE, Canada's Advanced Research and Innovation Network.

The project team would like to acknowledge and thank the many individuals and institutions that have contributed in one way or another to this platform. Your collaboration is extremely appreciated!

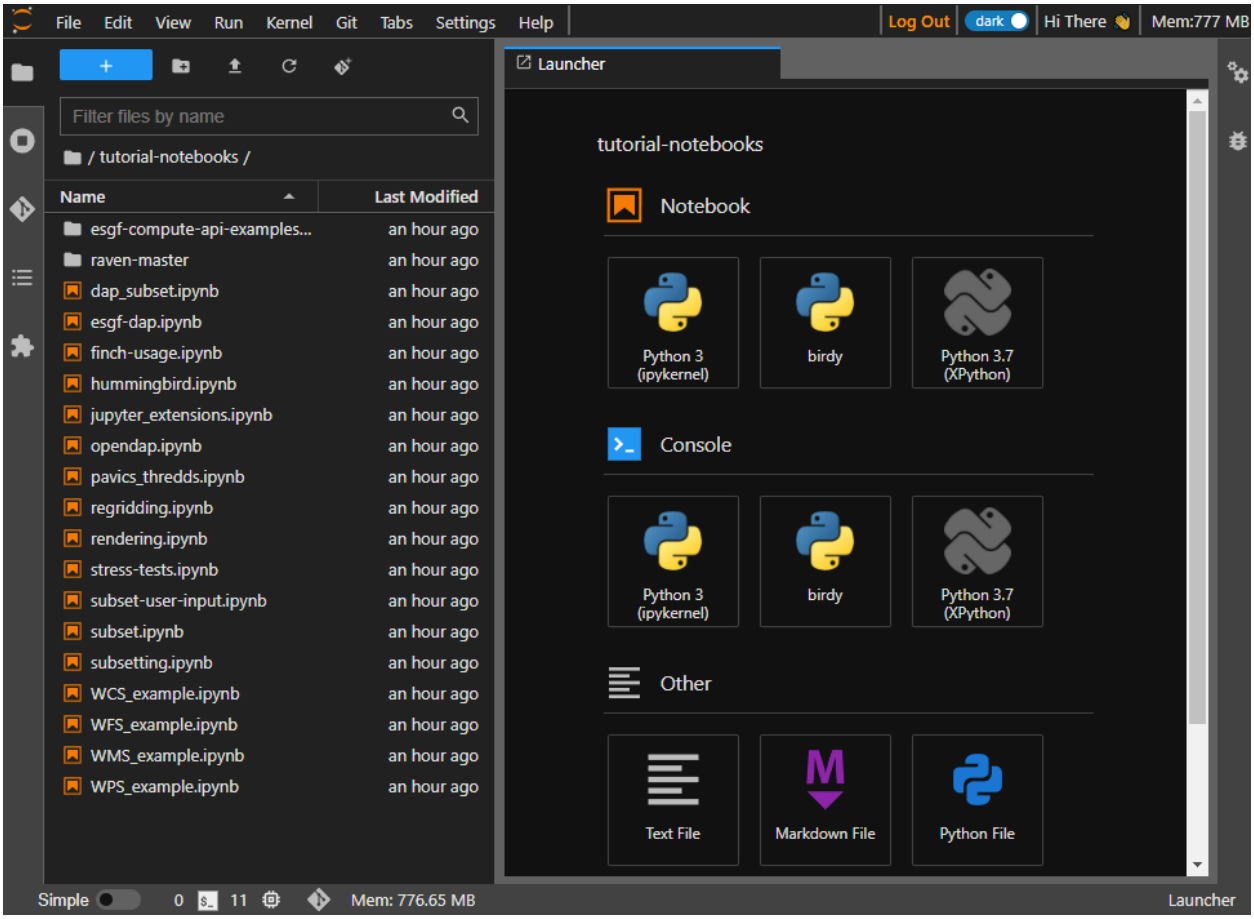
Getting started - Jupyter Notebook tutorial

PAVICS-Hydro can be accessed and used in multiple ways, but the most simple and straightforward method is to login to the PAVICS platform environment at:

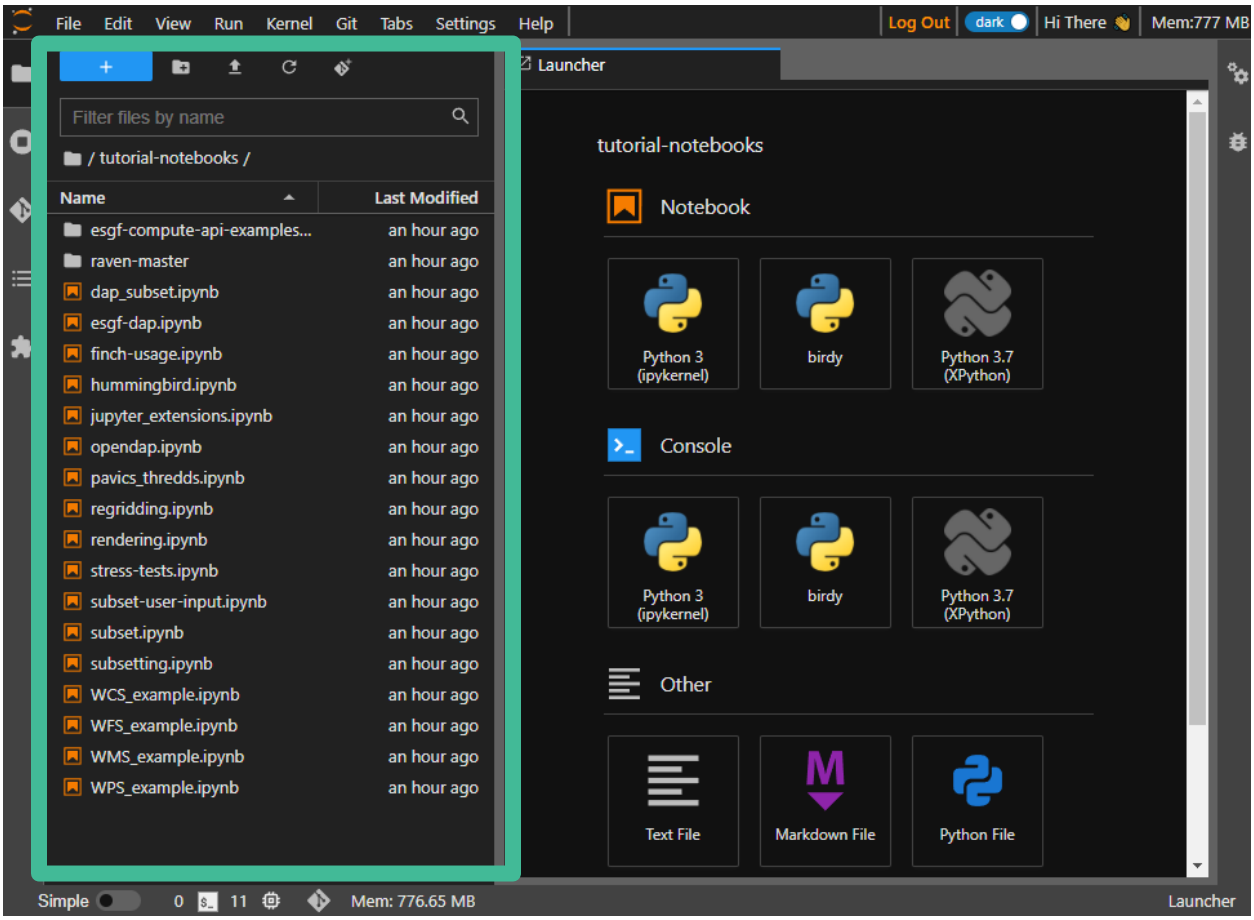
https://pavics.ouranos.ca/

You will need to login, or, if this is your first time accessing the platform, to register for an account by writing to pavics@ouranos.ca. This process will take approximately one day but is entirely free. The process is in place merely to ensure users can access a personalized workspace where they can store their files and data without having other users overwrite or delete them.

Once you have your account set up, simply login and your browser will bring you to a JupyterLab environment, which will look similar to this:



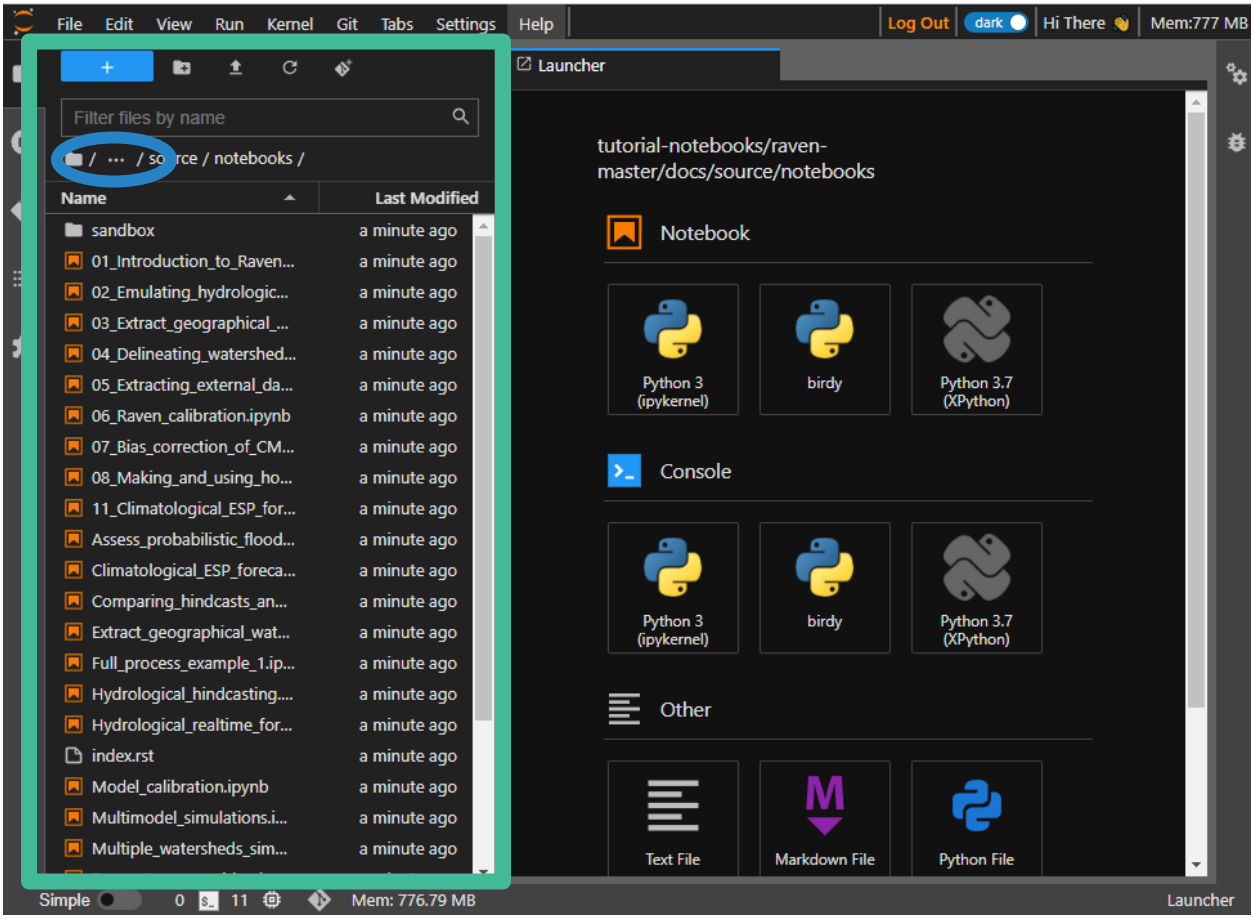
Let's first take the time to explore this environment. As you can see, the environment is split into 2 main panels. The leftmost panel (Green box) is the file explorer on the server.



By default, your account will login into the **/tutorial-notebooks/** directory. This is where many PAVICS tutorials can be found, including those related to climate change data processing. This is read-only workspace, so you cannot save changes in this location. We will start by focussing on the PAVICS-Hydro tools, but we encourage you to explore these notebooks as well at a later time.

Start by navigating to **/tutorial-notebooks/raven-master/docs/source/notebooks/**.

Here, you will find a series of notebooks that are specifically designed for PAVICS-Hydro. The notebooks starting with a number are the **tutorial notebooks**, which we will explore to start to get a feeling of what the platform can actually do.



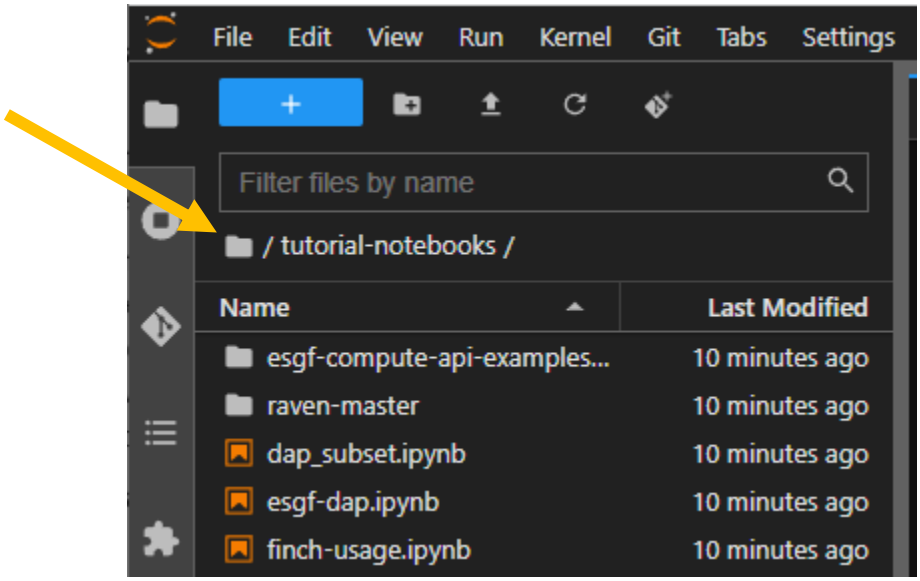
At this point, you could double-click the first tutorial notebook to explore it (**01_Introduction_to_Raven.ipynb)**, but the directory we are in right now is read-only. We therefore need to copy these files to our personal space on the server where data can be written to. This separation is to ensure that this folder remains up to date but will never overwrite your files during an update. Let's go ahead and do that now.

There are two ways to interact with files on the server:

1- Select the notebooks you want to copy, right-click, and copy. Then paste in the destination directory.
2- Select the notebooks you want to copy, right-click, and download. Then upload in the destination directory.

Use the method of your choice to copy the files, and then navigate to the root folder by clicking on the ellipsis circled in blue in the above figure until you are back to **/tutorial-notebooks/**.

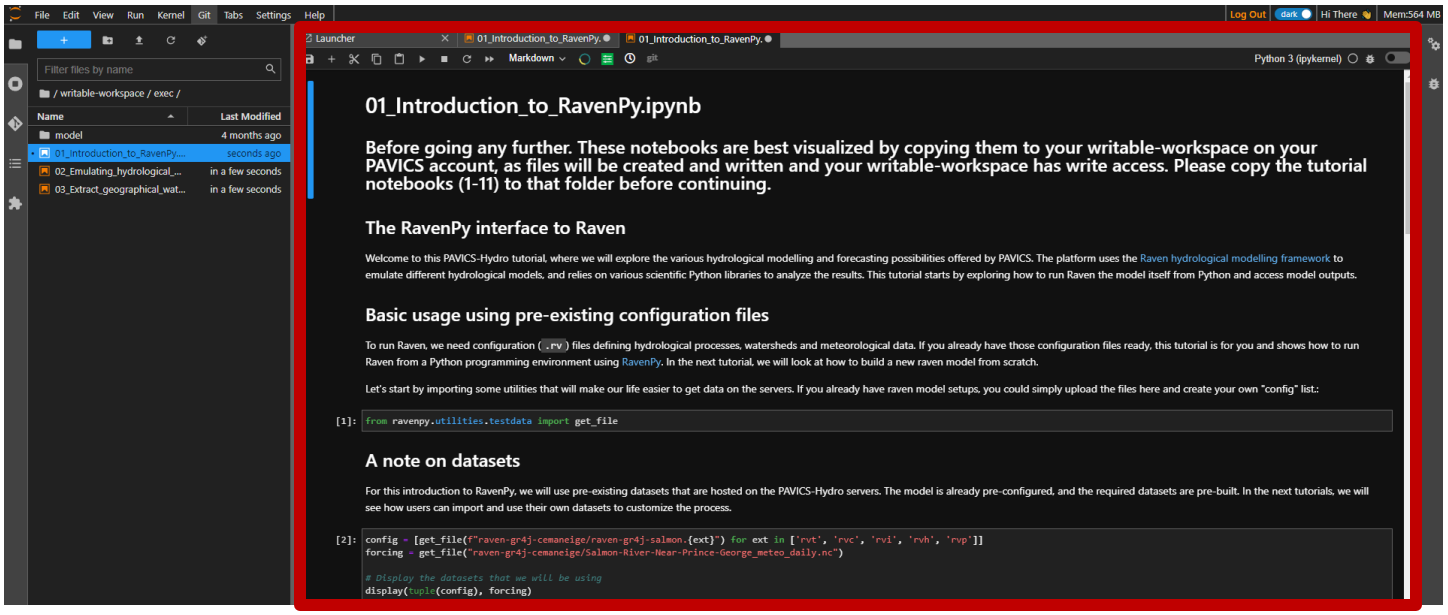
Now, click on the folder icon to the left of **/tutorial-notebooks/**, as shown here:



This will bring you to the root folder, where you now have access to your personalized directory **/writable-workspace/.** This will be our destination folder, so navigate to the **/writable-workspace/** now.

Your folder will probably be empty or have a few basic files when you first open it. However, all files you deposit or create in this folder will persist between sessions, meaning that they will not be overwritten or deleted even if you close your session and browser.

Right-click and paste the tutorial notebooks now. They will appear in your file explorer. You are now ready to open and run the first notebook! Go ahead and open the notebook **01_Introduction_to_Raven.ipynb** by double-clicking on the file. The right panel in your Jupyter environment should look like this (red rectangle):



There are a few things to notice here before running the code:

1- You can open multiple notebooks at once. They will open as tabs at the top of the right panel.

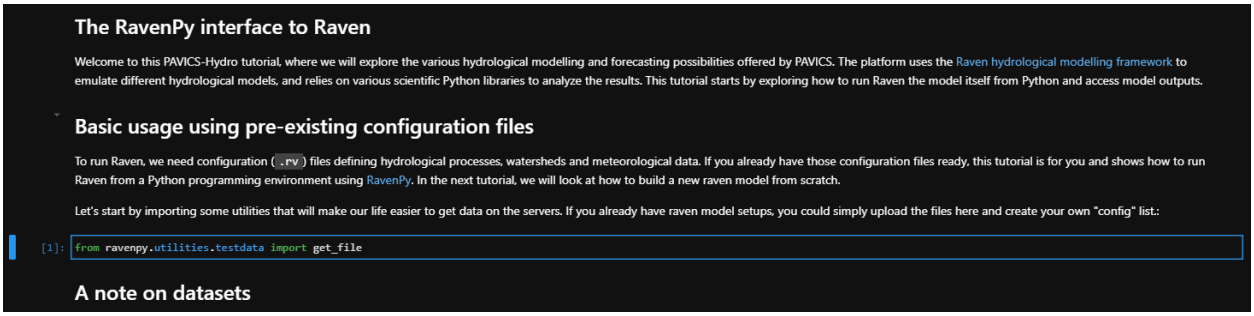
2- Every time a notebook is open, Jupyter will start to prepare an engine to run the code. This will take resources on the server, so we ask that you keep these to a minimum.

3- Even if you close a tab at the top of the screen, the underlying service is still running until you actively ask the service to shut down. You can see which notebooks are running by looking at the file explorer. A dot to the left of a filename indicates that the kernel is running. Please be considerate and shut down all kernels that you are not actively using (right-click on the file in the file explorer, and select "**Shut down kernel**"

4- You can see the total amount of memory your account is using at the top-right and bottom of the screen. The more kernels are open, and the more data is stored in memory, the higher the indicated value. Please try and keep these values as low as possible to ensure all users have a pleasant experience, especially when you log out of the platform. Yes, kernels are persistent and will continue running even if you log out of the platform, if you forget to close them!

With that in mind, we are ready to run the code!

Jupyter Notebooks (the *.ipynb* files displayed in the right panel) are interactive, which means that you can run parts of code, then wait for some results, then change the code and run it again, all in the same instance. We suggest that new users **select the first cell with code**, such that it is highlighted as follows:



The cell is selected, which means that when we ask Jupyter to run the code, it will run this cell specifically. Let's do that now, by clicking the "**Run the selected cells and advance**" button at the top of the panel, or by pressing "**Shift + Enter**". This will send the request to the server to run this line of code. We are starting by importing the packages we need, and the next cells all perform specific tasks such as preparing data, querying services to collect data or building a model structure, and so on. Read the text, and when you are done and ready to run the next cells, simply press the same button (or hit **"Shift + Enter"** again) to advance. You will see the code activating and processing when an asterisk appears next to the cell. When it has completed the calculation, the asterisk will change into a number, which is the number of cells that have been run in the current session. You can then move on to the next cell.

The tutorial notebooks themselves have many comments to guide users, so we will not go into much detail with respect to their content in this guide. Instead, follow the step-by-step approach detailed in the notebooks, starting with notebook 01, then 02, and advancing in numerical order.

These notebooks are updated on a regular basis and new ones will appear when new important features we want to share to new users become available.

This tutorial will give you a very good idea of the power and flexibility of PAVICS-Hydro and should suffice to be a solid starting point to adapt notebooks to your own needs. You can start from any of these notebooks, or you can create your own notebook from scratch or by recycling some pieces of code or cells.

The next documents, we will explore the specific features and functions in more depth which will allow fully unlocking the potential of PAVICS-Hydro.

For any questions or comments, please contact pavics@ouranos.ca.

Happy modelling!