# Hydropy: Python package for hydrological time series handling based on Python Pandas

Stijn Van Hoey (1), Sophie Balemans (1), Ingmar Nopens (1), Piet Seuntjens (2,3,4)

(1) Ghent University, Faculty of Bioscience Engineering, BIOMATH, Department of Mathematical Modelling, Statistics and Bioinformatics, Gent, Belgium (stvhoey.vanhoey@ugent.be), (2) Ghent University, Faculty of Bioscience Engineering, Department of Soil Management, Gent, Belgium (piet.seuntjens@vito.be), (3) Flemish Institute for Technological Research (VITO), Environmental modelling Unit, Mol, Belgium, (4) University of Antwerp, Department of Bioscience Engineering, Antwerp, Belgium

Most hydrologists are dealing with time series frequently. Reading in time series, transforming them and extracting specific periods for visualisation are part of the daily work. Spreadsheet software is used a lot for these operations, but has some major drawbacks. It is mostly not reproducible, it is prone to errors and not easy to automate, which results in repetitive work when dealing with large amounts of data. Scripting languages like R and Python on the other hand, provide flexibility, enable automation and reproducibility and, hence, increase efficiency.

Python has gained popularity over the last years and currently, tools for many aspects of scientific computing are readily available in Python. An increased support in controlling and managing the dependencies between packages (e.g. the Anaconda environment) allows for a wide audience to use the huge variety of available packages. Pandas is a powerful Python package for data analysis and has a lot of functionalities related to time series. As such, the package is of special interest to hydrologists. Some other packages, focussing on hydrology (e.g. Hydroclimpy by Pierre Gerard-Marchant and Hydropy by Javier Rovegno Campos), stopped active development, mainly due to the superior implementation of Pandas.

We present a (revised) version of the Hydropy package that is inspired by the aforementioned packages and builds on the power of Pandas. The main idea is to add hydrological domain knowledge to the already existing Pandas functionalities. Besides, the package attempts to make the time series handling intuitive and easy to perform, thus with a clear syntax.
Some illustrative examples of the current implementation starting from a Pandas DataFrame named flowdata:
Creating the object flow to work with:
*flow = HydroAnalysis(flowdata)*

Retrieve only the data during winter (across all years):
*flow.get_season('winter')*

Retrieve only the data during summer of 2010:
*flow.get_season('summer').get_year('2010')*
which is equivalent to
*flow.get_year('2010').get_season('summer')*

Retrieve only the data of July and get the peak values above the 95 percentile:
*flow.get_season('july').get_highpeaks(above_percentile=0.95)*

Retrieve only the data between two specified days and selecting only the rising limbs
*flow.get_date_range('01/10/2008', '15/2/2014').get_climbing()*

Calculate the annual sum and make a plot of it:
*flow.frequency_resample('A', 'sum').plot()*