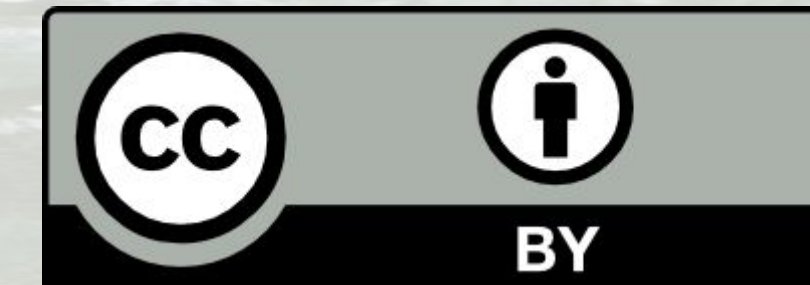


# Coupling Hydrological models using BMI in eWaterCycle



Niels Drost<sup>1</sup>, Rolf Hut<sup>2</sup>, Nick van De Giesen<sup>2</sup>, Ben van Werkhoven<sup>1</sup>, Jerom P. M. Aerts<sup>2</sup>, Jaro Camphuijsen<sup>1</sup>, Inti Pelupessy<sup>1</sup>, Berend Weel<sup>1</sup>, Stefan Verhoeven<sup>1</sup>, Ronald van Haren<sup>1</sup>, Eric Hutton<sup>3</sup>, Fakhreh Alidoost<sup>1</sup>, Gijs van Den Oord<sup>1</sup>, Yifat Dzigan<sup>1</sup>, Bouwe Andela<sup>1</sup>, Peter C. Kalverla<sup>1</sup>, and the Model Contributors\*

<sup>1</sup>Netherlands eScience Center, Amsterdam, Netherlands <sup>2</sup>Delft University of Technology, Faculty of Civil Engineering and Geosciences, Delft, Netherlands <sup>3</sup>Community Surface Dynamics Modeling System, Boulder, CO, United States



## Abstract

The eWaterCycle platform is a fully Open-Source platform built specifically to advance the state of FAIR and Open Science in hydrological Modeling. eWaterCycle builds on web technology, notebooks and containers to offer an integrated modeling experimentation environment for scientists. It allows scientists to run any supported hydrological model with ease, including setup and pre-processing of all data required. Common datasets such as ERA-Interim and ERA-5 forcing data and observations for verification of model output quality are available for usage by the models, and a Jupyter based interface is available for ease of use.

As the main API for models, we use the Basic Model Interface (BMI). This allows us to support models in a multitude of languages. Our gRPC based system (grpc4bmi) allows coupling of models, and running of multiple instances of the same model. Our system was designed to work with higher level interfaces such as PyMT, and we are currently integrating PyMT into our platform.

The BMI interface was specifically designed to make it easy to implement in any given model. During the FAIR Hydrological Modeling workshop a number of modelers worked on creating a BMI interface for their models, and making them available in the eWaterCycle system. We created a BMI interface for SUMMA, HYPE, MARRMoT, TopoFlex, LisFlood, WFLOW, and PCR-GLOBWB.

\*Model Contributors:

Louise Arnal, Fedor Baart, Andrew Bennett, Laurene Bouaziz, Marco Dal Molin, Fabrizio Fenicia, Emiliano Gelati, Caitlyn Hall, Jannis Hoch, Bert Jagers, Wouter Knoben, Lieke Melsen, Léonard Santos, Gharari Shervan, Edwin Sutanudjaja

## More Info

See more information in our website:

<https://www.ewatercycle.org>

See our code sources at GitHub:

<http://github.com/eWaterCycle>

Contact us at:

[ewatercycle@esciencecenter.nl](mailto:ewatercycle@esciencecenter.nl)

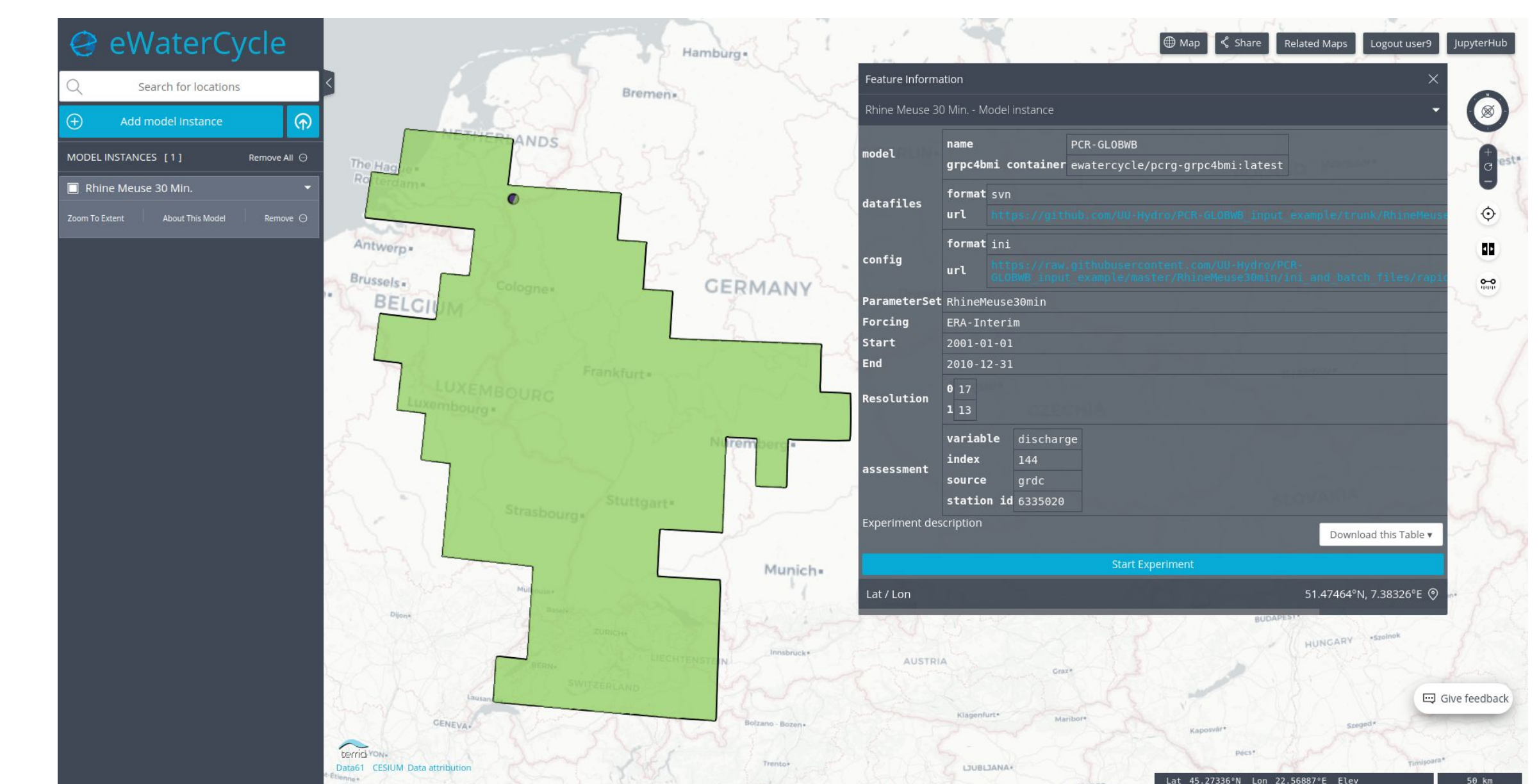
```
# create a model, in a container, connect using GRPC4BMI
model = BmiClientSingularity(
    image='ewatercycle-marmot-grpc4bmi.sif',
    input_dir='.', output_dir='.'
)
# initialize model, get some metadata
model.initialize(config_name)
time_units = model.get_time_units()

# run model, one step at a time, get some output
for current_time in time_range:
    model.update()
    value = model.get_value(variable)

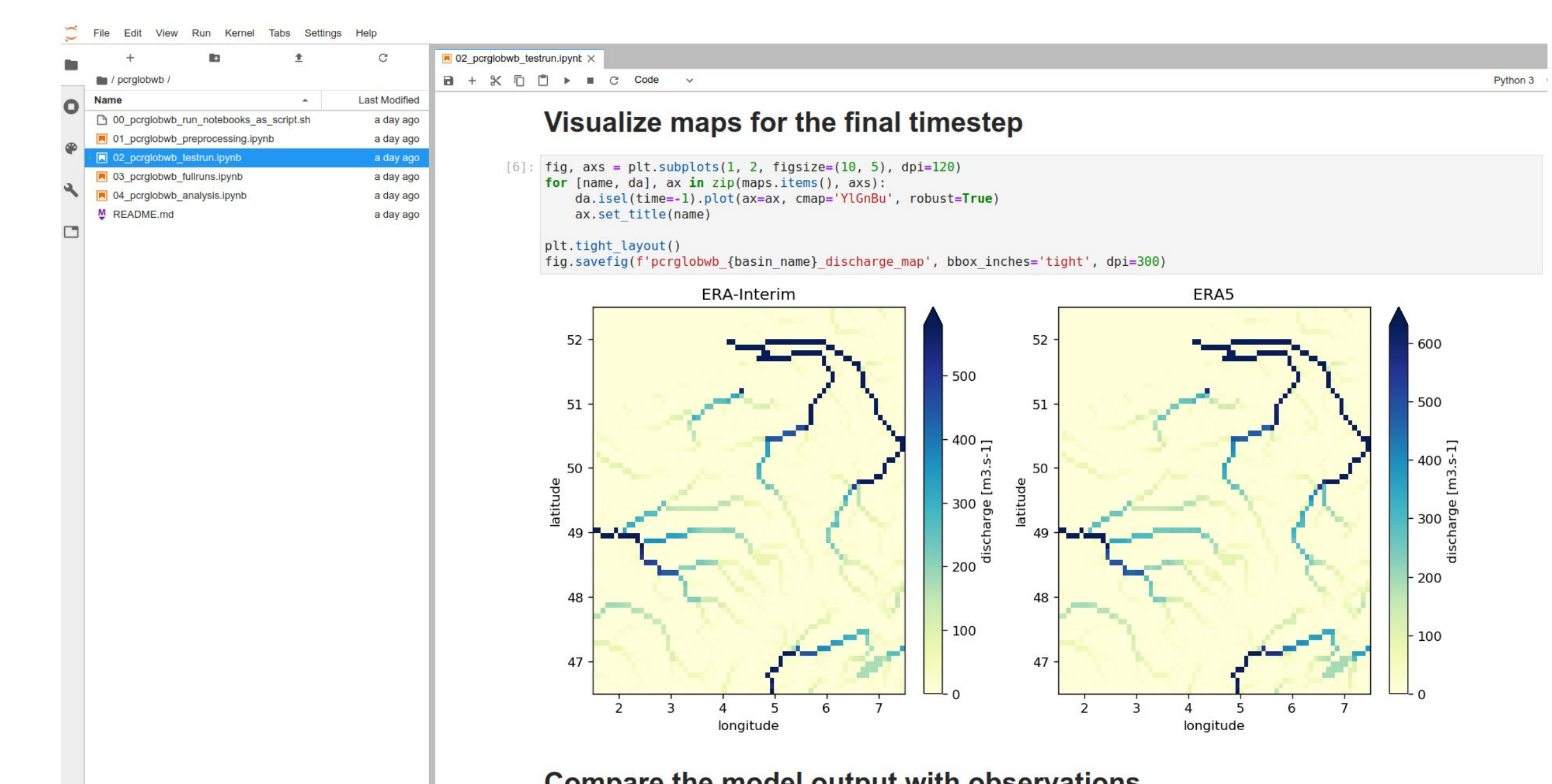
# clean up connection, container, client, etc
del model
```

Code snippet showing running a model (in this case [MARRMoT](#)) in a Jupyter notebook, using grpc4bmi and a Singularity container

## About eWaterCycle



eWaterCycle is a framework in which hydrological modelers can, for example, compare and analyze the results of models that use different sources of meteorological data. The goal of eWaterCycle is to advance the state of FAIR (Findable, Accessible, Interoperable, and Reusable) and open science in hydrological modeling.



The experiment runs in a Jupyter notebook, the model runs in a container, in any programming language, communicating through grpc4bmi, developed in our team.