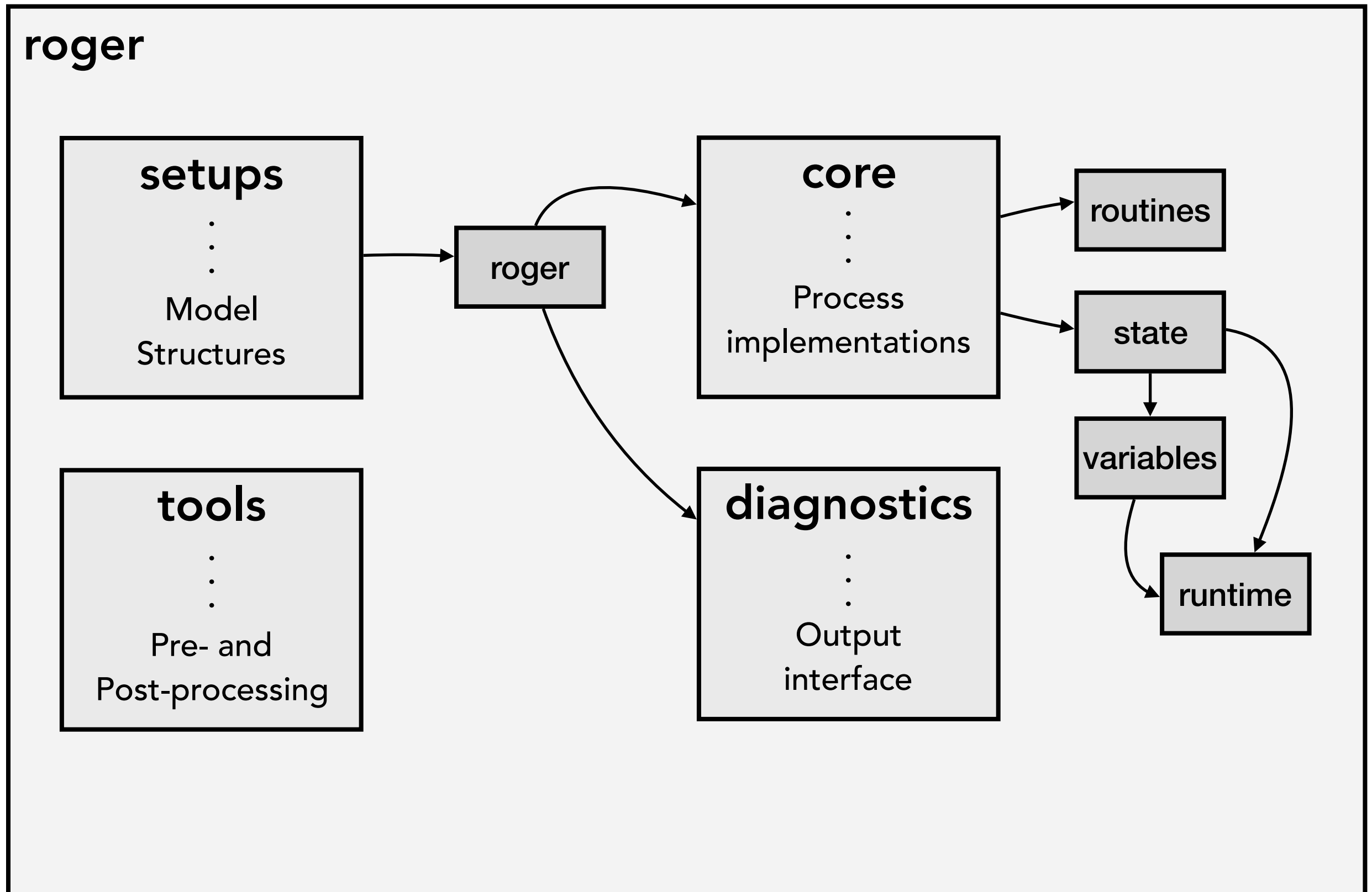


Runoff Generation Research (RoGeR) in pure Python

What's new?

- Community-Development based on a GitHub-Repository
- Online-Documentation
- Versioning and changelog
- Unit tests and continuous integration using GitHub Actions
- Flexible backends (numpy, jax, numpy-mpi, jax-mpi, jax-gpu)
- Tutorials
- netCDF for input and output
- New routines: Gravity-driven infiltration and percolation, groundwater, crop phenology/crop rotation and solute transport

Basic code structure - Python package *roger*



Code structure of main class for *roger*

RogerSetup

set_settings

Sets the model settings (e.g. number of rows)

set_grid

Sets the spatial and temporal grid (e.g. time increments)

set_look_up_tables

Sets the look-up tables (e.g. macropore flow velocities)

set_parameters

Sets the model parameters (e.g. soil depth)

set_initial_conditions

Sets the initial values (e.g. soil moisture)

set_forcing

Sets the model input (e.g. precipitation)

set_diagnostics

Sets the output variables (e.g. macropore infiltration)

after_timestep

Shifts time-dependent variables backward

***_setup**

Only called within setup

setup

Runs the model setup

run

Iterates over time steps

step

Calculates processes and updates storages

Example: 1D model

- Move to the directory containing script `oneD.py`

Windows:

Open *Command Prompt*

```
cd ...\roger\examples\plot_scale\oneD_tutorial
```

Mac:

Open *Terminal*

```
cd .../roger/examples/plot_scale/oneD_tutorial
```

- Activate Anaconda environment

Windows:

```
activate roger
```

Mac:

```
conda activate roger
```

- Run the model: `python oneD.py`
- Plot the model results: `python post_processing.py`

Example: 1D event model

- Move to the directory containing script `oneD_event.py`

Windows:

Open *Command Prompt*

```
cd ...\roger\examples\plot_scale\oneD_event_tutorial
```

Mac:

Open *Terminal*

```
cd .../roger/examples/plot_scale/oneD_event_tutorial
```

- Activate Anaconda environment

Windows:

```
activate roger
```

Mac:

```
conda activate roger
```

- Run the model: `python oneD_event.py`
- Plot the model results: `python post_processing.py`

TODO

- Routing routine
- Run-on infiltration routine
- Urban routine
- Write equations into docstrings
- Validate parameters and initial values beforehand
- Provide user-specific applications (e.g. .exe-files)