
WRF-SUEWS Documentation

Release v2020a

micromet@University of Reading, led by Prof Sue Grimmond

Nov 12, 2021

GETTING STARTED

1	Quick Start	3
1.1	Installation	3
1.2	Pre-processing	4
1.3	Simulation	5
1.4	Post-processing	5
2	Frequently Asked Questions	7
2.1	Where to download input data?	7
2.2	How to cite WRF-SUEWS?	7

Welcome to the WRF-SUEWS Documentation site!

QUICK START

1.1 Installation

1.1.1 Download source code

Clone WRF-SUEWS repository:

```
git clone git@github.com:Urban-Meteorology-Reading/WRF-SUEWS.git
```

After cloning the repo, make sure you use the following commands to update SUEWS repo associated with WRF-SUEWS

```
git submodule init
git submodule update
```

Note: An older version of SUEWS (v2018c) is used in this coupling work; the most up-to-date SUEWS is v2020a.

1.1.2 Couple WRF and SUEWS

Suppose you are in the root location of WRF-SUEWS directory, run the following code to set up a workspace folder for the coupled code (a folder named WRF-SUEWS by default; the name can be customised):

```
# go to the coupler directory
cd coupling-automator

# run the coupler
make
```

1.1.3 Compile WRF-SUEWS

Then go into the workspace folder (WRF-SUEWS if not set otherwise) and configure the compilation, which is now same as the standard WRF workflow:

```
./configure
```

Compile the code as follows:

```
./compile em_real >& compile.log
```

Note: If working on `jasmin`, you can also submit the job as following:

```
#!/bin/bash
#BSUB -q short-serial
#BSUB -o %J.out
#BSUB -e %J.err
#BSUB -W 02:30

./compile em_real >& log.compile
```

1.2 Pre-processing

The WRF-SUEWS pre-processing consists of two steps:

1.2.1 Standard WPS pre-processing

To generate the standard `wrfinput` files, please follow [the official tutorial](#). Once `wrfinput` (initial conditions) and `wrfbdy` (boundary conditions) files are generated by WPS, please further modify `wrfinput` using WRF-SUEWS pre-processor (WSPS) described as follows.

1.2.2 SUEWS-specific pre-processing

We strongly recommend [Anaconda-based python environment](#) for running WSPS.

After Anaconda is installed, please use `conda` to create a fresh environment named `WRF_SUEWS_pre` for WSPS:

```
cd input-processor/pre-processor-UK
conda env create -f environment.yml
```

Then switch to new environment and run WSPS:

```
conda activate WRF_SUEWS_pre
python3 wsps.py
```

Note: Please change the settings in `wsps.py` if necessary following instructions in `wsps`.

After running WSPS, WRF-SUEWS specific `wrfinput` netCDF files (e.g., `wrfinput_d03.suews`) will be generated under output.

1.3 Simulation

run simulation

1- After compilation of the code and pre-processing steps to prepare `wrf_input` files, you need to transfer all the `wrf_input` files to the location of main run (usually `[WRF-SUEWS directory]/test/em_real`). It should include the boundary condition file.

2- You also need to copy `namelist.suews` to the same location.

3- Use `LANDUSE.TBL` in `./test/em_real` to change the albedo associated with Urban areas (number 13 for `MODIFIED_IGBP_MODIS_NOAH` for both winter and summer. By default it is 15% (0.15). In London case, it is changed to 11%(0.11) based on Ward et al. 2016)

4- `namelist.input` should also be modified to be consistent for WRF-SUEWS. See examples [here](https://github.com/Urban-Meteorology-Reading/WRF-SUEWS/tree/master/input-processor/namelist_example/UK) (specially the `sf_surface_physics = 9` which specifies to use SUEWS as the LSM).

5- Finally, use the following script to run the simulations on JASMIN (go to `[WRF-SUEWS directory]/test/em_real`):

```
```#!/bin/bash #BSUB -q par-multi #BSUB -n 30 #BSUB -o %J.out #BSUB -e %J.err #BSUB -W 48:00
echo "Running WRF" mpirun ./wrf.exe ```
```

## 1.4 Post-processing



## FREQUENTLY ASKED QUESTIONS

### 2.1 Where to download input data?

will add...

### 2.2 How to cite WRF-SUEWS?

will add...