SUEWS-AnOHM

0.9

Generated by Doxygen 1.8.13

Contents

Chapter 1

README

Coding Guidelines http://urban-climate.net/umep/DevelopmentGuidelines#Coding_↩
Guidelines%7C

Manual http://urban-climate.net/umep/SUEWS

Actions needed

Benchmark data for testind developments

What is this repository for?

- Quick summary
- Version
- Learn Markdown

How do I get set up?

- Summary of set up
- Configuration
- Dependencies
- Database configuration
- How to run tests
- Deployment instructions

Contribution guidelines

- Writing tests
- Code review
- Other guidelines

Who do I talk to?

- Repo owner or admin
- Other community or team contact

Chapter 2

Modules Index

2.1 Modules List

Here is a list of all documented modules with brief descriptions:

Chapter 3

Data Type Index

3.1 Data Types List

Here are the data types with brief descriptions:

Chapter 4

Module Documentation

4.1 anohm_module Module Reference

AnOHM module: Analytical Objective Hysteresis Model.

Functions/Subroutines

- subroutine anohm (qn1, qn1_store, qn1_av_store, MetForcingData_grid, moist_surf, alb, emis, cpAnOHM, kkAnOHM, chAnOHM, sfr, nsurf, nsh, AnthropHeatMethod, id, a1, a2, a3, qs)

  High level wrapper for AnOHM calculation.
- subroutine **anohm_coef** (sfc_typ, xid, xgrid, MetForcingData_grid, moist_surf, AnthropHeatMethod, alb, emis, cpAnOHM, kkAnOHM, chAnOHM, xa1, xa2, xa3)
- subroutine **anohm_coef_land** (sfc_typ, xid, xgrid, MetForcingData_grid, AnthropHeatMethod, moist_surf, alb, emis, cpAnOHM, kkAnOHM, chAnOHM, xa1, xa2, xa3)
- subroutine **anohm_coef_land_cal** (ASd, mSd, ATa, mTa, tau, mWS, mWF, mAH, xalb, xemis, xcp, xk, xch, xBo, xa1, xa2, xa3)
- subroutine **anohm_tsurf_land_cal** (ASd, mSd, ATa, mTa, tau, mWS, mWF, mAH, xalb, xemis, xcp, xk, xch, xBo, ATs, mTs, gamma)
- subroutine **anohm_xts** (ASd, mSd, ATa, mTa, tau, mWS, mWF, mAH, xalb, xemis, xcp, xk, xch, xBo, tSd, xTHr, xTs)
- subroutine **anohm_coef_land_cal_extra** (ASd, mSd, ATa, mTa, tau, mWS, mWF, mAH, xalb, xemis, xcp, xk, xch, xBo, xa1, xa2, xa3, ATs, mTs, gamma)
- subroutine **anohm_coef_water** (sfc_typ, xid, xgrid, MetForcingData_grid, AnthropHeatMethod, moist_surf, alb, emis, cpAnOHM, kkAnOHM, chAnOHM, xa1, xa2, xa3)
- subroutine **anohm_coef_water_cal** (ASd, mSd, ATa, mTa, tau, mWS, mWF, mAH, xalb, xemis, xcp, xk, xch, xBo, xeta, xmu, xa1, xa2, xa3)
- subroutine **anohm_fc** (xid, MetForcingData_grid, AnthropHeatMethod, ASd, mSd, tSd, ATa, mTa, tTa, tau, mWS, mWF, mAH)
- subroutine **anohm_fcload** (xid, MetForcingData_grid, AnthropHeatMethod, Sd, Ta, RH, pres, WS, WF, AH, tHr)
- subroutine **anohm_fccal** (Sd, Ta, WS, WF, AH, tHr, ASd, mSd, tSd, ATa, mTa, tTa, tau, mWS, mWF, mAH)
- subroutine **anohm_shapefit** (tHr, obs, amp, mean, tpeak)
- subroutine **fsin** (m, n, x, xdat, ydat, fvec, iflag)
- subroutine **anohm_sfcload** (sfc_typ, xid, xgrid, alb, emis, cpAnOHM, kkAnOHM, chAnOHM, xalb, xemis, xcp, xk, xch, xBo)
- subroutine **anohm_bo_cal** (Ta, RH, pres, tHr, ASd, mSd, ATa, mTa, tau, mWS, mWF, mAH, xalb, xemis, xcp, xk, xch, xSM, tSd, xBo)
- subroutine **fcnbo** (n, x, fvec, iflag)
- real(kind(1d0)) function **qsat_fn** (Ta, pres)
- real(kind(1d0)) function **qa_fn** (Ta, RH, pres)

4.1.1 Detailed Description

AnOHM module: Analytical Objective Hysteresis Model.

Author

Ting Sun, ting.sun@reading.ac.uk

calculate heat storage. model details refer to https://doi.org/10.5194/gmd-2016-300

4.1.2 Function/Subroutine Documentation

4.1.2.1 anohm()

```
subroutine anohm_module::anohm (
             real(kind(1d0)), intent(in) qn1,
             real(kind(1d0)), dimension(nsh), intent(inout) qn1_store,
             real(kind(1d0)), dimension(2*nsh+1), intent(inout) qn1_av_store,
             real(kind(1d0)), dimension(:,:), intent(in) MetForcingData_grid,
             real(kind(1d0)), dimension(nsurf), intent(in) moist_surf,
             real(kind(1d0)), dimension(:), intent(in) alb,
             real(kind(1d0)), dimension(:), intent(in) emis,
             real(kind(1d0)), dimension(:), intent(in) cpAnOHM,
             real(kind(1d0)), dimension(:), intent(in) kkAnOHM,
             real(kind(1d0)), dimension(:), intent(in) chAnOHM,
             real(kind(1d0)), dimension(nsurf), intent(in) sfr,
             integer, intent(in) nsurf,
             integer, intent(in) nsh,
             integer, intent(in) AnthropHeatMethod,
             integer, intent(in) id,
             real(kind(1d0)), intent(out) a1,
             real(kind(1d0)), intent(out) a2,
             real(kind(1d0)), intent(out) a3,
             real(kind(1d0)), intent(out) qs )
```

High level wrapper for AnOHM calculation.

calculate heat storage based within AnOHM framework.

Parameters

this	object
x	an argument

Returns

1. grid ensemble heat storage: QS = a1∗(Q∗)+a2∗(dQ∗/dt)+a3
2. grid ensemble OHM coefficients: a1, a2 and a3

Chapter 5

Data Type Documentation

5.1 strings::value Interface Reference

Public Member Functions

- subroutine **value_dr** (str, rnum, ios)
- subroutine **value_sr** (str, rnum, ios)
- subroutine **value_di** (str, inum, ios)
- subroutine **value_si** (str, inum, ios)

The documentation for this interface was generated from the following file:

- stringmod.f95

5.2 ctrl_output::varattr Type Reference

Public Attributes

- character(len=15) **header**
- character(len=12) **unit**
- character(len=14) **fmt**
- character(len=50) **longnm**
- character(len=1) **aggreg**
- character(len=10) **group**
- integer **level**

The documentation for this type was generated from the following file:

- SUEWS_ctrl_output.f95

5.3 strings::writenum Interface Reference

Public Member Functions

- subroutine **write_dr** (rnum, str, fmt)
- subroutine **write_sr** (rnum, str, fmt)
- subroutine **write_di** (inum, str, fmt)
- subroutine **write_si** (inum, str, fmt)

The documentation for this interface was generated from the following file:

- stringmod.f95

5.4 strings::writeq Interface Reference

Public Member Functions

- subroutine **writeq_dr** (unit, namestr, value, fmt)
- subroutine **writeq_sr** (unit, namestr, value, fmt)
- subroutine **writeq_di** (unit, namestr, ivalue, fmt)
- subroutine **writeq_si** (unit, namestr, ivalue, fmt)

The documentation for this interface was generated from the following file:

- stringmod.f95

Index