MOSSCO

0.0.1a

Generated by Doxygen 1.8.3.1

Wed Jun 12 2013 13:30:38

Contents

Chapter 1

Data Type Index

1.1 Data Types List

Here are the data types with brief descriptions:

Chapter 2

File Index

2.1 File List

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

Chapter 3

Data Type Documentation

3.1 esmf_fabm_sediment_component Module Reference

Public Member Functions

- subroutine, public **empty_setservices** (gridcomp, rc)

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

- esmf_fabm_sediment_component.F90

3.2 fabm_sediment_driver Module Reference

The FABM sediment driver module provides infrastructure for the MOSSCO sediment component. The driver provides tendencies for state variables as sum of local rates (through FABM) and vertical diffusion.

Public Member Functions

- subroutine, public init_sed_grid (grid)

  Initialise sediment grid.
- subroutine, public init_fabm_sed (sed)

  Initialise FABM sediment driver.
- subroutine, public fabm_sed_get_rhs (sed, bdys, fluxes, rhs)

  get right-hand sides
- subroutine, public finalize_fabm_sed ()

  finalize the FABM sediment driver

3.2.1 Detailed Description

The FABM sediment driver module provides infrastructure for the MOSSCO sediment component. The driver provides tendencies for state variables as sum of local rates (through FABM) and vertical diffusion.

3.2.2 Member Function/Subroutine Documentation

3.2.2.1 subroutine, public fabm_sediment_driver::fabm_sed_get_rhs (type(type_sed), intent(inout) *sed,* real(rk), dimension(1:_inum_,1:_jnum_,1:sed%nvar+1), intent(in) *bdys,* real(rk), dimension(1:_inum_,1:_jnum_,1:sed%nvar), intent(inout) *fluxes,* real(rk), dimension(1:_inum_,1:_jnum_,1:_knum_,1:sed%nvar), intent(out) *rhs*)

get right-hand sides

The right-hand sides for integration are provided for the state variables. The local tendencies are provided through FABM, the local changes due to diffusion are calculated in diff3d. Boundary conditions handled through the diffusion routine, where particulate properties use a flux boundary condition and dissolved properties use a concentration boundary condition. Diffusivities are calculated here depending on temperature (first index in bdys vector)

3.2.2.2 subroutine, public fabm_sediment_driver::finalize_fabm_sed ( )

finalize the FABM sediment driver

deallocate all the arrays

3.2.2.3 subroutine, public fabm_sediment_driver::init_fabm_sed (type(type_sed), intent(inout) *sed*)

Initialise FABM sediment driver.

Assumes to have a grid, either created by e.g. init_sed_grid. Parameters are read from namelist sed_nml, FABM is initialised and necessary arrays are allocated. Porosity is set here.

3.2.2.4 subroutine, public fabm_sediment_driver::init_sed_grid (type(fabm_sed_grid), intent(inout) *grid*)

Initialise sediment grid.

Allocate memory, create a grid and fill the sed_grid_type. The number of layers is set outside in beforehand by the sediment component.

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

- fabm_sediment_driver.F90

Chapter 4

File Documentation

4.1 fabm_sediment_driver.F90 File Reference

FABM sediment driver.

```
#include "fabm_driver.h"
```

Data Types

- module fabm_sediment_driver

  The FABM sediment driver module provides infrastructure for the MOSSCO sediment component. The driver provides tendencies for state variables as sum of local rates (through FABM) and vertical diffusion.

Macros

- #define **_GRID_** sed%grid
- #define **_INUM_** _GRID_%inum
- #define **_JNUM_** _GRID_%jnum
- #define **_KNUM_** _GRID_%knum

4.1.1 Detailed Description

FABM sediment driver. The driver contains the sediment driver module

Author

Richard Hofmeister

Index