MOHIDLagrangian

0.01

Generated by Doxygen 1.8.14

Contents

Chapter 1

Modules Index

1.1 Modules List

Here is a list of all modules with brief descriptions:

Chapter 2

Data Type Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Data Type Index

3.1 Data Types List

Here are the data types with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

Module Documentation

5.1 commom_modules Module Reference

Module to hold all of the commonly used base modules.

5.1.1 Detailed Description

Module to hold all of the commonly used base modules.

Author

Ricardo Birjukovs Canelas

5.2 geometry Module Reference

Module that defines geometry classes and related methods.

Data Types

- type box
- type line

  Type - point class.
- type point
- type shape
- type sphere

5.2.1 Detailed Description

Module that defines geometry classes and related methods.

Author

Ricardo Birjukovs Canelas

5.3 initialize Module Reference

Functions/Subroutines

- subroutine, public initmohidlagrangian (xmlfilename)

  Birjukovs Canelas - MARETEC

5.3.1 Function/Subroutine Documentation

5.3.1.1 initmohidlagrangian()

```
subroutine, public initialize::initmohidlagrangian (
          type(string), intent(in) xmlfilename )
```

Birjukovs Canelas - MARETEC

Public xml parser routine. Builds the simulation space from the input xml case file.

Parameters

in	*xmlfilename*	

5.4 simulation_parameters Module Reference

Module to hold all the simulation related parameters, definitions and methods.

Functions/Subroutines

- subroutine, public setsimparameter (parmkey, parmvalue)

  Birjukovs Canelas - MARETEC
- subroutine, public setsimgravity (grav)

  Birjukovs Canelas - MARETEC
- subroutine, public setsimrho (read_rho)

  Birjukovs Canelas - MARETEC
- subroutine, public setsimdp (read_dp)

  Birjukovs Canelas - MARETEC
- subroutine, public setsimbounds (point_, coords)

  Birjukovs Canelas - MARETEC

Variables

- integer, public integrator = 1
- real(prec), public cfl = 0.5
- real(prec), public initfreeze = 0.0
- real(prec), public timemax = MV
- real(prec), public timeout = MV
- real(prec), public dp = MV
- type(vector), public pointmin
- type(vector), public pointmax
- type(vector), public gravity
- real(prec), public rho_ref = 1000.0

5.4.1 Detailed Description

Module to hold all the simulation related parameters, definitions and methods.

Author

Ricardo Birjukovs Canelas

5.4.2 Function/Subroutine Documentation

5.4.2.1 setsimbounds()

```
subroutine, public simulation_parameters::setsimbounds (
            type(string), intent(in) point_,
            type(vector) coords )
```

Birjukovs Canelas - MARETEC

Public bounding box setting routine.

Parameters

in	*point_,coords*	

5.4.2.2 setsimdp()

```
subroutine, public simulation_parameters::setsimdp (
            type(string), intent(in) read_dp )
```

Birjukovs Canelas - MARETEC

Public dp setting routine.

Parameters

in	*read_dp*	

5.4.2.3 setsimgravity()

```
subroutine, public simulation_parameters::setsimgravity (
            type(vector) grav )
```

Birjukovs Canelas - MARETEC

Public Gravity setting routine.

Parameters

in	*grav*	

5.4.2.4 setsimparameter()

```
subroutine, public simulation_parameters::setsimparameter (
            type(string), intent(in) parmkey,
            type(string), intent(in) parmvalue )
```

Birjukovs Canelas - MARETEC

Public parameter setting routine. Builds the simulation parametric space from the input xml case file.

Parameters

in	*parmkey*	
in	*parmvalue*	

5.4.2.5 setsimrho()

```
subroutine, public simulation_parameters::setsimrho (
            type(string), intent(in) read_rho )
```

Birjukovs Canelas - MARETEC

Public Rho_Ref setting routine.

Parameters

in	*rho*	

5.4.3 Variable Documentation

5.4.3.1 cfl

```
real(prec), public simulation_parameters::cfl = 0.5
```

5.4.3.2 dp

```
real(prec), public simulation_parameters::dp = MV
```

5.4.3.3 gravity

```
type(vector), public simulation_parameters::gravity
```

5.4.3.4 initfreeze

```
real(prec), public simulation_parameters::initfreeze = 0.0
```

5.4.3.5 integrator

```
integer, public simulation_parameters::integrator = 1
```

5.4.3.6 pointmax

```
type(vector), public simulation_parameters::pointmax
```

5.4.3.7 pointmin

```
type(vector), public simulation_parameters::pointmin
```

5.4.3.8 rho_ref

```
real(prec), public simulation_parameters::rho_ref = 1000.0
```

5.4.3.9 timemax

```
real(prec), public simulation_parameters::timemax = MV
```

5.4.3.10 timeout

```
real(prec), public simulation_parameters::timeout = MV
```

5.5 source Module Reference

Module to hold and wrap all the tracer sources respective modules. Defines a source class and related methods.

5.5.1 Detailed Description

Module to hold and wrap all the tracer sources respective modules. Defines a source class and related methods.

Author

Ricardo Birjukovs Canelas

5.6 source_emitter Module Reference

Module that defines a source geometry class, emitter class and related methods.

5.6.1 Detailed Description

Module that defines a source geometry class, emitter class and related methods.

Author

Ricardo Birjukovs Canelas

5.7 source_identity Module Reference

Module that defines a source class and related methods.

Functions/Subroutines

- subroutine, public allocsources (nsources)

  Birjukovs Canelas - MARETEC
- subroutine, public initsource (num, id, name, emitting_rate, source_geometry, geometry)

  Birjukovs Canelas - MARETEC

Variables

- type(source_class), dimension(:), allocatable, public source

5.7.1 Detailed Description

Module that defines a source class and related methods.

Author

Ricardo Birjukovs Canelas

5.7.2 Function/Subroutine Documentation

5.7.2.1 allocsources()

```
subroutine, public source_identity::allocsources (
              integer, intent(in) nsources )
```

Birjukovs Canelas - MARETEC

source allocation routine - allocates the sources objects

Parameters

in	*nsources*	

5.7.2.2 initsource()

```
subroutine, public source_identity::initsource (
              integer, intent(in) num,
              integer, intent(in) id,
              type(string), intent(in) name,
              real(prec), intent(in) emitting_rate,
              type(string), intent(in) source_geometry,
              class(shape), intent(in) geometry )
```

Birjukovs Canelas - MARETEC

source inititialization routine - Generates a source and initializes its variables

Parameters

out	*source*	
in	*num,id,name,emitting_rate,source_geometry*	

5.7.3 Variable Documentation

5.7.3.1 source

```
type(source_class), dimension(:), allocatable, public source_identity::source
```

5.8 tracer Module Reference

Module to hold and wrap all the tracer respective modules. Defines a pure Lagrangian tracer class. This is intended to serve as the base class for every type of tracer class needed, that should be
built as derived of this class, with the necessary modifiers to model the desired behaviour. Basic tracer data (parameters, variables) are implemented. Tracer methods such as I/O, integration and interpolation routines are implemented.

5.8.1 Detailed Description

Module to hold and wrap all the tracer respective modules. Defines a pure Lagrangian tracer class. This is intended to serve as the base class for every type of tracer class needed, that should be
built as derived of this class, with the necessary modifiers to model the desired behaviour. Basic tracer data (parameters, variables) are implemented. Tracer methods such as I/O, integration and interpolation routines are implemented.

Author

Ricardo Birjukovs Canelas

5.9 tracer2d Module Reference

Module that defines a pure Lagrangian 2D tracer class and related methods, as a subset of the tracer3D module.

Functions/Subroutines

- subroutine tracer2d_init (trc, filename, time, x, is_sigma)

  Birjukovs Canelas - MARETEC Routine Author Name and Affiliation.

5.9.1 Detailed Description

Module that defines a pure Lagrangian 2D tracer class and related methods, as a subset of the tracer3D module.

Author

Ricardo Birjukovs Canelas

5.9.2 Function/Subroutine Documentation

5.9.2.1 tracer2d_init()

```
subroutine tracer2d::tracer2d_init (
            type(tracer_class), intent(out) trc,
            character(len=*), intent(in) filename,
            real(prec_time) time,
            real(prec), dimension(:), intent(in) x,
            logical, intent(in) is_sigma )
```

Birjukovs Canelas - MARETEC Routine Author Name and Affiliation.

Brief description of routine.

2D Tracer inititialization routine - Generates a tracer collection and initializes their variables

Parameters

out	*trc*	
in	*filename*	

5.10 tracer3d Module Reference

Module that defines a pure Lagrangian tracer class and related methods.

Functions/Subroutines

- subroutine, public tracer_init (trc, id, time, x, y, z)

  Birjukovs Canelas - MARETEC

5.10.1 Detailed Description

Module that defines a pure Lagrangian tracer class and related methods.

Author

Ricardo Birjukovs Canelas

5.10.2 Function/Subroutine Documentation

5.10.2.1 tracer_init()

```
subroutine, public tracer3d::tracer_init (
            type(tracer_class), intent(inout) trc,
            integer, intent(in) id,
            real(prec_time), intent(in) time,
            real(prec), intent(in) x,
            real(prec), intent(in) y,
            real(prec), intent(in) z )
```

Birjukovs Canelas - MARETEC

Tracer inititialization routine - Generates a tracer and initializes its variables

Parameters

out	*trc*	
in	*filename*	

5.11 tracer_interp Module Reference

5.12 tracer_precision Module Reference

Module to control the precision of the variables trough the project.

Variables

- integer, parameter, public prec = sp
- integer, parameter, public prec_time = sp
- integer, parameter, public prec_wrt = sp
- real(prec), parameter, public missing_value_default = -9999.0_dp
- real(prec), parameter, public mv = MISSING_VALUE_DEFAULT
- real(prec), parameter, public mv_int = int(MISSING_VALUE_DEFAULT)
- real(prec), parameter, public err_dist = 1E8_dp
- integer, parameter, public err_ind = -1

5.12.1 Detailed Description

Module to control the precision of the variables trough the project.

Author

Ricardo Birjukovs Canelas

5.12.2 Variable Documentation

5.12.2.1 err_dist

```
real(prec), parameter, public tracer_precision::err_dist = 1E8_dp
```

5.12.2.2 err_ind

```
integer, parameter, public tracer_precision::err_ind = -1
```

5.12.2.3 missing_value_default

```
real(prec), parameter, public tracer_precision::missing_value_default = -9999.0_dp
```

5.12.2.4 mv

```
real(prec), parameter, public tracer_precision::mv = MISSING_VALUE_DEFAULT
```

5.12.2.5 mv_int

```
real(prec), parameter, public tracer_precision::mv_int = int(MISSING_VALUE_DEFAULT)
```

5.12.2.6 prec

```
integer, parameter, public tracer_precision::prec = sp
```

5.12.2.7 prec_time

```
integer, parameter, public tracer_precision::prec_time = sp
```

5.12.2.8 prec_wrt

```
integer, parameter, public tracer_precision::prec_wrt = sp
```

Chapter 6

Data Type Documentation

6.1 geometry::box Type Reference

Inheritance diagram for geometry::box:

6.2 geometry::line Type Reference

Type - point class.

Inheritance diagram for geometry::line:

Collaboration diagram for geometry::line:

Public Attributes

- type(vector) last

  Type - line class.
- type(vector) coordinates
- type(vector) of
- type(vector) the
- type(vector) end
- type(vector) point

6.2.1 Detailed Description

Type - point class.

6.2.2 Member Data Documentation

6.2.2.1 coordinates

```
type(vector) geometry::line::coordinates
```

6.2.2.2 end

```
type(vector) geometry::line::end
```

6.2.2.3 last

```
type(vector) geometry::line::last
```

Type - line class.

6.2.2.4 of

```
type(vector) geometry::line::of
```

6.2.2.5 point

```
type(vector) geometry::line::point
```

6.2.2.6 the

```
type(vector) geometry::line::the
```

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

- C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/geometry.f90

6.3 geometry::point Type Reference

Inheritance diagram for geometry::point:

Collaboration diagram for geometry::point:

Additional Inherited Members

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

- C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/geometry.f90

6.4 geometry::shape Type Reference

Inheritance diagram for geometry::shape:

Public Attributes

- type(vector) pt

  Type - extendable shape class.
- type(vector) coordinates
- type(vector) of
- type(vector) a
- type(vector) point

6.4.1 Member Data Documentation

6.4.1.1 a

```
type(vector) geometry::shape::a
```

6.4.1.2 coordinates

```
type(vector) geometry::shape::coordinates
```

6.4.1.3 of

```
type(vector) geometry::shape::of
```

6.4.1.4 point

```
type(vector) geometry::shape::point
```

6.4.1.5 pt

```
type(vector) geometry::shape::pt
```

Type - extendable shape class.

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

- C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/geometry.f90

6.5 geometry::sphere Type Reference

Inheritance diagram for geometry::sphere:

Collaboration diagram for geometry::sphere:

Public Attributes

- real(prec) radius

  Type - sphere class.
- real(prec) sphere

6.5.1 Member Data Documentation

6.5.1.1 radius

```
real(prec) geometry::sphere::radius
```

Type - sphere class.

6.5.1.2 sphere

```
real(prec) geometry::sphere::sphere
```

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

- C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/geometry.f90

Chapter 7

File Documentation

7.1 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/app/main.f90 File Reference

7.2 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/common_↩ modules.f90 File Reference

Modules

- module commom_modules

  Module to hold all of the commonly used base modules.

7.3 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/geometry.f90 File Reference

Data Types

- type geometry::shape
- type geometry::point
- type geometry::line

  Type - point class.
- type geometry::sphere
- type geometry::box

Modules

- module geometry

  Module that defines geometry classes and related methods.

7.4 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/initialize.f90 File Reference

Modules

- module initialize

Functions/Subroutines

- subroutine, public initialize::initmohidlagrangian (xmlfilename)

    Birjukovs Canelas - MARETEC

7.5 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/simulation_↩ parameters.f90 File Reference

Modules

- module simulation_parameters

    Module to hold all the simulation related parameters, definitions and methods.

Functions/Subroutines

- subroutine, public simulation_parameters::setsimparameter (parmkey, parmvalue)

    Birjukovs Canelas - MARETEC
- subroutine, public simulation_parameters::setsimgravity (grav)

    Birjukovs Canelas - MARETEC
- subroutine, public simulation_parameters::setsimrho (read_rho)

    Birjukovs Canelas - MARETEC
- subroutine, public simulation_parameters::setsimdp (read_dp)

    Birjukovs Canelas - MARETEC
- subroutine, public simulation_parameters::setsimbounds (point_, coords)

    Birjukovs Canelas - MARETEC

Variables

- integer, public simulation_parameters::integrator = 1
- real(prec), public simulation_parameters::cfl = 0.5
- real(prec), public simulation_parameters::initfreeze = 0.0
- real(prec), public simulation_parameters::timemax = MV
- real(prec), public simulation_parameters::timeout = MV
- real(prec), public simulation_parameters::dp = MV
- type(vector), public simulation_parameters::pointmin
- type(vector), public simulation_parameters::pointmax
- type(vector), public simulation_parameters::gravity
- real(prec), public simulation_parameters::rho_ref = 1000.0

7.6 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/source.f90 File Reference

Modules

- module source

  Module to hold and wrap all the tracer sources respective modules. Defines a source class and related methods.

7.7 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/source_↩ emitter.f90 File Reference

Modules

- module source_emitter

  Module that defines a source geometry class, emitter class and related methods.

7.8 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/source_↩ identity.f90 File Reference

Modules

- module source_identity

  Module that defines a source class and related methods.

Functions/Subroutines

- subroutine, public source_identity::allocsources (nsources)

  Birjukovs Canelas - MARETEC
- subroutine, public source_identity::initsource (num, id, name, emitting_rate, source_geometry, geometry)

  Birjukovs Canelas - MARETEC

Variables

- type(source_class), dimension(:), allocatable, public source_identity::source

7.9 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/tracer.f90 File Reference

Modules

- module tracer

  Module to hold and wrap all the tracer respective modules. Defines a pure Lagrangian tracer class. This is intended to serve as the base class for every type of tracer class needed, that should be built as derived of this class, with the necessary modifiers to model the desired behaviour. Basic tracer data (parameters, variables) are implemented. Tracer methods such as I/O, integration and interpolation routines are implemented.

7.10 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/tracer2D.f90 File Reference

Modules

- module tracer2d

  Module that defines a pure Lagrangian 2D tracer class and related methods, as a subset of the tracer3D module.

Functions/Subroutines

- subroutine tracer2d::tracer2d_init (trc, filename, time, x, is_sigma)

  Birjukovs Canelas - MARETEC Routine Author Name and Affiliation.

7.11 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/tracer3D.f90 File Reference

Modules

- module tracer3d

  Module that defines a pure Lagrangian tracer class and related methods.

Functions/Subroutines

- subroutine, public tracer3d::tracer_init (trc, id, time, x, y, z)

  Birjukovs Canelas - MARETEC

7.12 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/tracer_↩ interp.f90 File Reference

Modules

- module tracer_interp

7.13 C:/Users/administrator/Documents/GitHub/MOHID-Lagrangian/src/lib/tracer_↩ precision.f90 File Reference

Modules

- module tracer_precision

  Module to control the precision of the variables trough the project.

Variables

- integer, parameter, public tracer_precision::prec = sp
- integer, parameter, public tracer_precision::prec_time = sp
- integer, parameter, public tracer_precision::prec_wrt = sp
- real(prec), parameter, public tracer_precision::missing_value_default = -9999.0_dp
- real(prec), parameter, public tracer_precision::mv = MISSING_VALUE_DEFAULT
- real(prec), parameter, public tracer_precision::mv_int = int(MISSING_VALUE_DEFAULT)
- real(prec), parameter, public tracer_precision::err_dist = 1E8_dp
- integer, parameter, public tracer_precision::err_ind = -1

Index