TOY API

Generated by Doxygen 1.8.11

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

Namespace Documentation

4.1 LONG Namespace Reference

LONG implementation.

Classes

- class Collision
  Collision class.

4.1.1 Detailed Description

LONG implementation.

4.2 MARIO Namespace Reference

Mario implementation.

Classes

- class Nbody

4.2.1 Detailed Description

Mario implementation.

4.3 NB Namespace Reference

namespace for NB module

Classes

- class Nbody

  a class for defining a particle
- class Parameter

  parameter class shared by all particles

Functions

- class NB::Parameter evolution (Particle ∗_particle, const Int _n, const PertParticle ∗_perturber, const PertForce ∗_pert_force, const Float _time_start, const Float _time_end, Float _time_real, Parameter &_ext_pars)

  evolution function

4.3.1 Detailed Description

namespace for NB module

4.3.2 Function Documentation

4.3.2.1 class NB::Parameter NB::evolution (Particle ∗ *_particle,* const Int *_n,* const PertParticle ∗ *_perturber,* const PertForce ∗ *_pert_force,* const Float *_time_start,* const Float *_time_end,* Float *_time_real,* Parameter & *_ext_pars*)

evolution function

Parameters

in	*_particle*	: particle set (the c.m. particle, the subset is linked in the branches)
in	*_n*	: number of particles
in	*_perturber*	perturbers for external interaction of the sub-group (read only)
in	*_pert_force*	perturber force for prediction, or external force (e.g. PN, tidal force) depending on the implementation
in	*_time_start*	: starting time of evolution
in	*_time_end*	expected time of ending evolution
out	*_time_real*	the real evolved time from the evolution function
in,out	*_ext_pars*	extra parameters needed for the N-body code (some global parameter used for all particles or control parameters) depend on implementation

Chapter 5

Class Documentation

5.1 LONG::Collision Class Reference

Collision class.

`#include <collision.h>`

Public Member Functions

- void call (const double _r12)
  check collision

5.1.1 Detailed Description

Collision class.

5.1.2 Member Function Documentation

5.1.2.1 void LONG::Collision::call (const double *_r12*)

check collision

Parameters

in	*_r12*	distance between 1 and 2

The documentation for this class was generated from the following files:

- collision.h
- collision.cpp

5.2 MARIO::Nbody Class Reference

`#include <nbody.h>`

Public Member Functions

- void evolution (const double _time_end, double &_time_real)
  evolution function

5.2.1 Member Function Documentation

5.2.1.1 void MARIO::Nbody::evolution (const double *_time_end*, double & *_time_real*)

evolution function

evolution function for Nbody

Parameters

in	*_time_end*	expected time of ending evolution
out	*_time_real*	the real evolved time from the evolution function

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- nbody.h
- nbody.cpp

5.3 NB::Nbody Class Reference

a class for defining a particle

`#include <sample.h>`

Public Member Functions

- Particle (const Float _mass, const FloatVector &_pos, const FloatVector &_vel)

  constructor
- Float getMass () const

  Get mass (required for ARC::chain)
- const FloatVector getPos () const

  Get position.
- const FloatVector getVel () const

  Get velocity.
- void setPos (const FloatVector &_pos)

  Set position.
- void setVel (const FloatVector &_vel)

  Set velocity.
- void setMass (const Float m)

  Set mass (required for ARC::chain)
- Particle * getBranch1 ()

  branch 1 address for binary tree
- Particle * getBranch2 ()

  branch 2 address for binary tree

5.3.1 Detailed Description

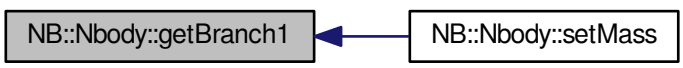
a class for defining a particle

5.3.2 Member Function Documentation

5.3.2.1 Particle* NB::Nbody::getBranch1 ( )

branch 1 address for binary tree

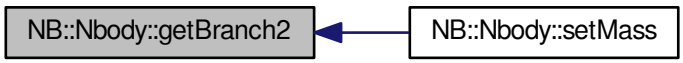
Here is the caller graph for this function:



5.3.2.2 Particle* NB::Nbody::getBranch2 ( )

branch 2 address for binary tree

Here is the caller graph for this function:



5.3.2.3 Float NB::Nbody::getMass ( ) const [inline]

Get mass (required for ARC::chain)

Returns

mass

5.3.2.4 const FloatVector NB::Nbody::getPos ( ) const [inline]

Get position.

Returns

position vector

5.3.2.5 const FloatVector NB::Nbody::getVel ( ) const [inline]

Get velocity.

Returns

velocity vector

5.3.2.6 NB::Nbody::Particle (const Float *_mass,* const FloatVector & *_pos,* const FloatVector & *_vel*) [inline]

constructor

Parameters

in	*_mass*	particle mass
in	*_pos*	position
in	*_vel*	velocity

Here is the caller graph for this function:

5.3.2.7 void NB::Nbody::setMass (const Float *m*) [inline]

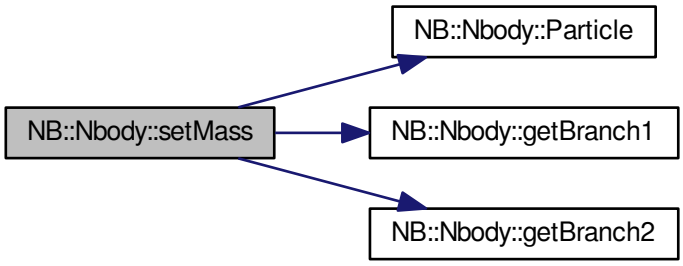
Set mass (required for ARC::chain)

NAN check will be done

Parameters

in	*m*	particle mass

Here is the call graph for this function:



5.3.2.8 void NB::Nbody::setPos (const FloatVector & *_pos*) [inline]

Set position.

Parameters

in	*_pos*	position vector

5.3.2.9 void NB::Nbody::setVel (const FloatVector & *_vel*) [inline]

Set velocity.

Parameters

in	*_vel*	velocity vector

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

- sample.h

5.4 NB::Parameter Class Reference

parameter class shared by all particles

`#include <sample.h>`

5.4.1 Detailed Description

parameter class shared by all particles

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

- sample.h

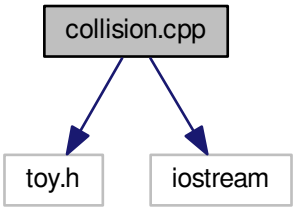
Chapter 6

File Documentation

6.1 collision.cpp File Reference

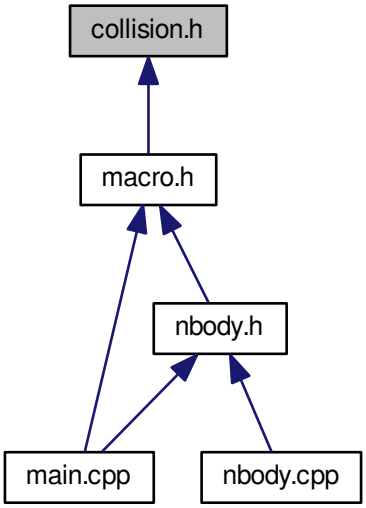
```
#include <toy.h>
#include <iostream>
```

Include dependency graph for collision.cpp:



6.2 collision.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

- class LONG::Collision

  Collision class.

Namespaces

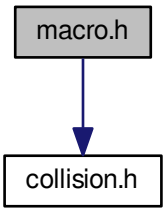
- LONG

  LONG implementation.

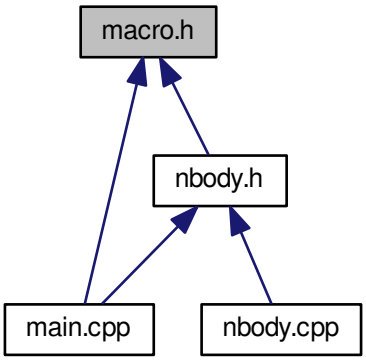
6.3 macro.h File Reference

```
#include <collision.h>
```

Include dependency graph for macro.h:



This graph shows which files directly or indirectly include this file:



Macros

- #define __Collision LONG::Collision

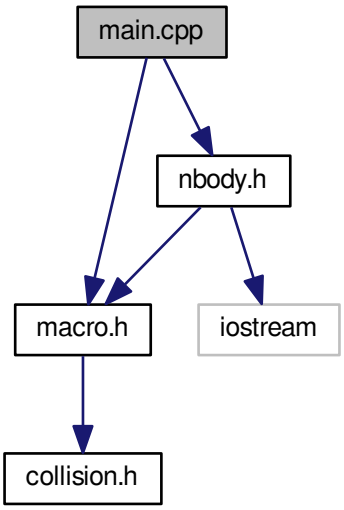
6.3.1 Macro Definition Documentation

6.3.1.1 #define __Collision LONG::Collision

6.4 main.cpp File Reference

```
#include <macro.h>
#include <nbody.h>
```

Include dependency graph for main.cpp:



Functions

- int main ()

  Main function.

6.4.1 Function Documentation

6.4.1.1 int main ( )

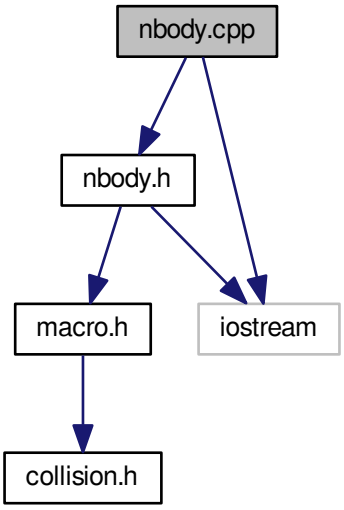
Main function.

Here is the call graph for this function:

6.5 nbody.cpp File Reference

```
#include <nbody.h>
#include <iostream>
```

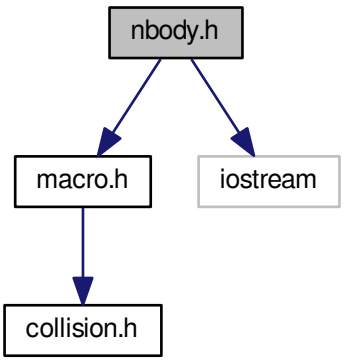
Include dependency graph for nbody.cpp:



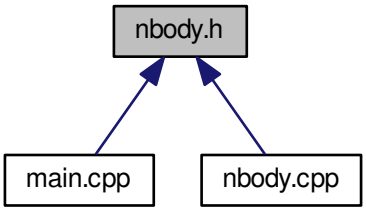
6.6 nbody.h File Reference

```
#include <macro.h>
#include <iostream>
```

Include dependency graph for nbody.h:



This graph shows which files directly or indirectly include this file:



Classes

- class MARIO::Nbody

Namespaces

- MARIO

  Mario implementation.

6.7 sample.h File Reference

Classes

- class NB::Nbody

  a class for defining a particle
- class NB::Parameter

  parameter class shared by all particles

Namespaces

- NB

  namespace for NB module

Functions

- class NB::Parameter NB::evolution (Particle *_particle, const Int _n, const PertParticle *_perturber, const PertForce *_pert_force, const Float _time_start, const Float _time_end, Float _time_real, Parameter &_ext↩ _pars)

  evolution function

Index