

Módulo de Física - RKE

Generated by Doxygen 1.8.0

Fri May 18 2012 09:26:58

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	struct_objeto Struct Reference	5
3.1.1	Detailed Description	5
3.2	struct_vetor Struct Reference	5
3.2.1	Detailed Description	6
4	File Documentation	7
4.1	include/rkefisica.h File Reference	7
4.1.1	Detailed Description	7
4.1.2	Function Documentation	7
4.1.2.1	rke_adiciona_objeto	7
4.1.2.2	rke_get_objeto	8
4.1.2.3	rke_set_arrasto	8
4.1.2.4	rke_set_delta_t	8
4.1.2.5	rke_set_numero_objetos	8
4.1.2.6	rke_set_vetor_mundo	8
4.1.2.7	rke_simula	8
4.2	include/rketypes.h File Reference	8
4.2.1	Detailed Description	9
4.2.2	Typedef Documentation	9
4.2.2.1	objeto	9
4.2.2.2	vetor	9
4.3	src/fisica.c File Reference	9
4.3.1	Detailed Description	10
4.3.2	Function Documentation	10
4.3.2.1	itera_posicao	10

4.3.2.2	rke_adiciona_objeto	10
4.3.2.3	rke_get_objeto	11
4.3.2.4	rke_set_arrasto	11
4.3.2.5	rke_set_delta_t	11
4.3.2.6	rke_set_numero_objetos	11
4.3.2.7	rke_set_vetor_mundo	11
4.3.2.8	rke_simula	12
4.4	src/main.c File Reference	12
4.4.1	Detailed Description	12

Chapter 1

Class Index

1.1 Class List

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

struct_objeto	5
struct_vetor	5

Chapter 2

File Index

2.1 File List

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

include/ rkefisica.h	Arquivo header da biblioteca de funções físicas	7
include/ rketypes.h	Arquivo header de tipos e defines do Red Knife Engine	8
src/ fisica.c	Esta é a biblioteca de funções que lidam com a física do Red Knife Engine	9
src/ main.c	Este é o arquivo que implementa as funções descritas na fisica.c . Aqui, carrega-se um arquivo texto com as condições iniciais e escreve um arquivo "saida.out" com as informações após as iterações	12

Chapter 3

Class Documentation

3.1 struct_objeto Struct Reference

```
#include <rketypes.h>
```

Public Attributes

- int **id**
- double **x**
- double **y**
- double **v_x**
- double **v_y**
- double **massa**
- double **tempo**

3.1.1 Detailed Description

Struct objeto

Parameters

<i>id</i>	Identificador único
<i>x</i>	Componente x
<i>y</i>	Componente y
<i>v_x</i>	Componente x da velocidade do objeto
<i>v_y</i>	Componente y da velocidade do objeto
<i>massa</i>	Massa do objeto
<i>tempo</i>	Tempo de vida do objeto

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

- [include/rketypes.h](#)

3.2 struct_vetor Struct Reference

```
#include <rketypes.h>
```

Public Attributes

- double **x**
- double **y**

3.2.1 Detailed Description

Struct vetor

Parameters

<i>x</i>	Componente x
<i>y</i>	Componente y

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

- include/[rketypes.h](#)

Chapter 4

File Documentation

4.1 include/rkefisica.h File Reference

Arquivo header da biblioteca de funções físicas.

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

Functions

- void [rke_set_delta_t](#) (double d_t)
- void [rke_set_arrasto](#) (double coef_arrasto)
- void [rke_set_vetor_mundo](#) (double x, double y)
- void [rke_set_numero_objetos](#) (int numero)
- void [rke_adiciona_objeto](#) (int id, double x, double y, double v_x, double v_y, double massa, double tempo)
- [objeto rke_get_objeto](#) (int i)
- void [rke_simula](#) ()

4.1.1 Detailed Description

Arquivo header da biblioteca de funções físicas.

Author

João da Silva, Marina Salles, Ricardo Macedo

4.1.2 Function Documentation

4.1.2.1 void [rke_adiciona_objeto](#) (int *id*, double *x*, double *y*, double *v_x*, double *v_y*, double *massa*, double *tempo*)

Adiciona um objeto à lista de objetos a serem simulados.

Parameters

<i>id</i>	Identificador único do objeto
<i>x</i>	Posição x do objeto
<i>y</i>	Posição y do objeto
<i>v_x</i>	Velocidade em x do objeto
<i>v_y</i>	Velocidade em y do objeto
<i>massa</i>	Massa do objeto
<i>tempo</i>	Tempo de vida do objeto em segundos

4.1.2.2 objeto rke_get_objeto (int *i*)

Retorna o *i*-ésimo objeto.

Parameters

<i>i</i>	Índice do objeto
----------	------------------

4.1.2.3 void rke_set_arrasto (double *coef_arrasto*)

Configura o coeficiente de arrasto da superfície.

Parameters

<i>coef_arrasto</i>	Coeficiente de 0.0 a 1.0
---------------------	--------------------------

4.1.2.4 void rke_set_delta_t (double *d_t*)

Indica a resolução da simulação. Este é o tamanho do quanta de tempo.

Parameters

<i>d_t</i>	Resolução em segundos
------------	-----------------------

4.1.2.5 void rke_set_numero_objetos (int *numero*)

Configura o número total de objetos a serem simulados.

Parameters

<i>numero</i>	Número de objetos
---------------	-------------------

4.1.2.6 void rke_set_vetor_mundo (double *x*, double *y*)

Configura o vetor base que registrará todos os objetos do mundo.

Parameters

<i>x</i>	Componente x
<i>y</i>	Componente y

4.1.2.7 void rke_simula ()

Simula todos os objetos por um quanta de tempo.

4.2 include/rketypes.h File Reference

Arquivo header de tipos e defines do Red Knife Engine.

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

Classes

- struct [struct_vetor](#)
- struct [struct_objeto](#)

Defines

- #define **BARCOID** -1
- #define **ESTATICO** -1.0

Typedefs

- typedef struct [struct_vetor](#) [vetor](#)
- typedef struct [struct_objeto](#) [objeto](#)

4.2.1 Detailed Description

Arquivo header de tipos e defines do Red Knife Engine.

Author

João da Silva, Marina Salles, Ricardo Macedo

4.2.2 Typedef Documentation

4.2.2.1 typedef struct [struct_objeto](#) [objeto](#)

Struct objeto

Parameters

<i>id</i>	Identificador único
<i>x</i>	Componente x
<i>y</i>	Componente y
<i>v_x</i>	Componente x da velocidade do objeto
<i>v_y</i>	Componente y da velocidade do objeto
<i>massa</i>	Massa do objeto
<i>tempo</i>	Tempo de vida do objeto

4.2.2.2 typedef struct [struct_vetor](#) [vetor](#)

Struct vetor

Parameters

<i>x</i>	Componente x
<i>y</i>	Componente y

4.3 src/fisica.c File Reference

Esta é a biblioteca de funções que lidam com a física do Red Knife Engine.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../include/rketypes.h"
Include dependency graph for fisica.c:
```

Functions

- void [rke_set_delta_t](#) (double d_t)
- void [rke_set_arrasto](#) (double coef_arrasto)
- void [rke_set_vetor_mundo](#) (double x, double y)
- void [rke_set_numero_objetos](#) (int numero)
- void [itera_posicao](#) ([objeto](#) *obj, [vetor](#) forca)
- void [rke_adiciona_objeto](#) (int id, double x, double y, double v_x, double v_y, double massa, double tempo)
- [objeto](#) [rke_get_objeto](#) (int i)
- void [rke_simula](#) ()

Variables

- double **delta_t**
- double **percentual_atrito**
- double **arrasto**
- [vetor](#) **mundo**
- int **ult_objeto** = 0
- int **num_objetos** = 0
- [objeto](#) * **objetos**

4.3.1 Detailed Description

Esta é a biblioteca de funções que lidam com a física do Red Knife Engine.

Author

João da Silva, Marina Salles, Ricardo Macedo

4.3.2 Function Documentation

4.3.2.1 void itera_posicao ([objeto](#) * *obj*, [vetor](#) *forca*)

De acordo com a velocidade do objeto, o tempo de vida dele, o vetor mundo e a massa, calcula a posição seguinte no próximo quanta de tempo.

Parameters

<i>obj</i>	Endereço do objeto
<i>forca</i>	Força a ser aplicada

4.3.2.2 void rke_adiciona_objeto (int *id*, double *x*, double *y*, double *v_x*, double *v_y*, double *massa*, double *tempo*)

Adiciona um objeto à lista de objetos a serem simulados.

Parameters

<i>id</i>	Identificador único do objeto
<i>x</i>	Posição x do objeto
<i>y</i>	Posição y do objeto
<i>v_x</i>	Velocidade em x do objeto
<i>v_y</i>	Velocidade em y do objeto
<i>massa</i>	Massa do objeto
<i>tempo</i>	Tempo de vida do objeto em segundos

4.3.2.3 objeto rke_get_objeto (int *i*)

Retorna o i-ésimo objeto.

Parameters

<i>i</i>	Índice do objeto
----------	------------------

4.3.2.4 void rke_set_arrasto (double *coef_arrasto*)

Configura o coeficiente de arrasto da superfície.

Parameters

<i>coef_arrasto</i>	Coeficiente de 0.0 a 1.0
---------------------	--------------------------

4.3.2.5 void rke_set_delta_t (double *d_t*)

Indica a resolução da simulação. Este é o tamanho do quanta de tempo.

Parameters

<i>d_t</i>	Resolução em segundos
------------	-----------------------

4.3.2.6 void rke_set_numero_objetos (int *numero*)

Configura o número total de objetos a serem simulados.

Parameters

<i>numero</i>	Número de objetos
---------------	-------------------

4.3.2.7 void rke_set_vetor_mundo (double *x*, double *y*)

Configura o vetor base que registrará todos os objetos do mundo.

Parameters

<i>x</i>	Componente x
<i>y</i>	Componente y

4.3.2.8 void rke_simula ()

Simula todos os objetos por um quanta de tempo.

4.4 src/main.c File Reference

Este é o arquivo que implementa as funções descritas na [fisica.c](#). Aqui, carrega-se um arquivo texto com as condições iniciais e escreve um arquivo "saida.out" com as informações após as iterações.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "../include/rketypes.h"
#include "../include/rkefisica.h"
Include dependency graph for main.c:
```

Functions

- int **main** (int argc, char *argv[])

4.4.1 Detailed Description

Este é o arquivo que implementa as funções descritas na [fisica.c](#). Aqui, carrega-se um arquivo texto com as condições iniciais e escreve um arquivo "saida.out" com as informações após as iterações.

Author

João da Silva, Marina Salles, Ricardo Macedo

Index

fisica.c

- itera_posicao, [10](#)
- rke_adiciona_objeto, [10](#)
- rke_get_objeto, [11](#)
- rke_set_arrasto, [11](#)
- rke_set_delta_t, [11](#)
- rke_set_numero_objetos, [11](#)
- rke_set_vetor_mundo, [11](#)
- rke_simula, [11](#)

include/rkefisica.h, [7](#)

include/rketypes.h, [8](#)

itera_posicao

fisica.c, [10](#)

objeto

rketypes.h, [9](#)

rke_adiciona_objeto

fisica.c, [10](#)

rkefisica.h, [7](#)

rke_get_objeto

fisica.c, [11](#)

rkefisica.h, [7](#)

rke_set_arrasto

fisica.c, [11](#)

rkefisica.h, [8](#)

rke_set_delta_t

fisica.c, [11](#)

rkefisica.h, [8](#)

rke_set_numero_objetos

fisica.c, [11](#)

rkefisica.h, [8](#)

rke_set_vetor_mundo

fisica.c, [11](#)

rkefisica.h, [8](#)

rke_simula

fisica.c, [11](#)

rkefisica.h, [8](#)

rkefisica.h

rke_adiciona_objeto, [7](#)

rke_get_objeto, [7](#)

rke_set_arrasto, [8](#)

rke_set_delta_t, [8](#)

rke_set_numero_objetos, [8](#)

rke_set_vetor_mundo, [8](#)

rke_simula, [8](#)

rketypes.h

objeto, [9](#)

vetor, [9](#)

src/fisica.c, [9](#)

src/main.c, [12](#)

struct_objeto, [5](#)

struct_vetor, [5](#)

vetor

rketypes.h, [9](#)