

EF2:Peers_EI

0.1

Generated by Doxygen 1.8.0

Tue Apr 24 2012 21:35:59

Contents

1	Desarrolladores EF2	1
1.1	Ricardo Prato	1
1.2	Catalina Dominguez	1
2	Changes in EF2:Peers_EI Version 1.1	3
3	Class Index	5
3.1	Class Hierarchy	5
4	Class Index	7
4.1	Class List	7
5	File Index	9
5.1	File List	9
6	Class Documentation	11
6.1	cell Class Reference	11
6.1.1	Detailed Description	11
6.2	char Class Reference	11
6.2.1	Detailed Description	11
6.3	colvec Class Reference	12
6.3.1	Detailed Description	12
6.4	double Class Reference	12
6.4.1	Detailed Description	12
6.5	function_handle Class Reference	12
6.5.1	Detailed Description	12
6.6	handle Class Reference	13
6.6.1	Detailed Description	13
6.6.2	Member Data Documentation	14
6.6.2.1	addlistener	14

6.6.2.2	delete	14
6.6.2.3	disp	14
6.6.2.4	display	14
6.6.2.5	eq	14
6.6.2.6	fieldnames	14
6.6.2.7	fields	15
6.6.2.8	findobj	15
6.6.2.9	findprop	15
6.6.2.10	isvalid	15
6.6.2.11	notify	15
6.6.2.12	permute	15
6.6.2.13	reshape	15
6.6.2.14	sort	15
6.6.2.15	transpose	15
6.7	integer Class Reference	16
6.7.1	Detailed Description	16
6.8	logical Class Reference	16
6.8.1	Detailed Description	16
6.9	matrix Class Reference	16
6.9.1	Detailed Description	17
6.10	rowvec Class Reference	17
6.10.1	Detailed Description	17
6.11	struct Class Reference	17
6.11.1	Detailed Description	17
6.12	varargin Class Reference	17
6.12.1	Detailed Description	18
6.13	varargout Class Reference	18
6.13.1	Detailed Description	18
7	File Documentation	19
7.1	auxiliar.m File Reference	19
7.1.1	Function Documentation	19
7.1.1.1	genbasftn	19
7.2	Bases_func_Maxwell.m File Reference	19
7.2.1	Detailed Description	20
7.2.2	Function Documentation	20
7.2.2.1	Bases_func_Maxwell	20

7.3	carst_basftn.m File Reference	21
7.3.1	Function Documentation	21
7.3.1.1	carst_basftn	21
7.3.1.2	mtoc_subst_carst_basftn_m_tsbus_cotm_test_line	21
7.4	class_substitutes.c File Reference	21
7.5	developers.c File Reference	21
7.6	funciones.m File Reference	22
7.6.1	Function Documentation	22
7.6.1.1	funciones	22
7.7	genbasftn.m File Reference	22
7.7.1	Function Documentation	22
7.7.1.1	genbasftn	22
7.8	inside_triangle.m File Reference	22
7.8.1	Detailed Description	22
7.8.2	Function Documentation	23
7.8.2.1	inside_triangle	23
7.9	intextnodes.m File Reference	23
7.9.1	Function Documentation	24
7.9.1.1	intextnodes	24
7.9.1.2	mtoc_subst_intextnodes_m_tsbus_cotm_invM	24
7.10	main.m File Reference	24
7.10.1	Detailed Description	24
7.10.2	Function Documentation	24
7.10.2.1	main	24
7.11	mallao1.m File Reference	25
7.11.1	Function Documentation	25
7.11.1.1	mallao1	25
7.12	provideGeometricData.m File Reference	25
7.12.1	Function Documentation	25
7.12.1.1	provideGeometricData	25
7.13	refineRGB.m File Reference	25
7.13.1	Detailed Description	26
7.13.2	Function Documentation	26
7.13.2.1	refineRGB	26
7.14	shownu.m File Reference	26
7.14.1	Function Documentation	26
7.14.1.1	shownu	27

Chapter 1

Desarrolladores EF2

1.1 Ricardo Prato

<https://sites.google.com/site/clasesrp/> -Profesor tiempo completo -Universidad del Norte

1.2 Catalina Dominguez

<https://sites.google.com/site/clasesrp/>

-Profesora tiempo completo -Universidad del Norte

Chapter 2

Changes in EF2:Peers_EI Version 1.1

Member [Bases_func_Maxwell](#) (matlabtypesubstitute u, matlabtypesubstitute i, matlabtypesubstitute j, matlabtypesubstitute x)

([Catalina Dominguez](#), 2012-04-16)

Chapter 3

Class Index

3.1 Class Hierarchy

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

cell	11
char	11
colvec	12
double	12
function_handle	12
handle	13
integer	16
logical	16
matrix	16
sparsematrix	??
rowvec	17
struct	17
varargin	17
varargout	18

Chapter 4

Class Index

4.1 Class List

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

cell	A MatLab cell array or matrix	11
char	A MatLab character array	11
colvec	A matlab column vector	12
double	A double value	12
function_handle	A MatLab function handle	12
handle	Matlab's base handle class (documentation generation substitute)	13
integer	An integer value	16
logical	A boolean value	16
matrix	A matlab matrix	16
rowvec	A matlab row vector	17
sparsematrix	A matlab sparse matrix	??
struct	A MatLab struct	17
varargin	A variable number of input arguments	17
varargout	A variable number of output arguments	18

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

auxiliar.m	19
Bases_func_Maxwell.m	
Calcula el valor en el punto x de la función base ϕ_j^i (elementos de Whitney 2D) definido por	
$\phi_1^i = \lambda_1 \cdot \nabla \lambda_2 - \lambda_2 \cdot \nabla \lambda_1$	
$\phi_2^i = \lambda_1 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_1$	
$\phi_3^i = \lambda_2 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_2$	
	19
carst_basftn.m	21
class_substitutes.c	21
developers.c	21
funciones.m	22
genbasftn.m	22
inside_triangle.m	
Inside_triangle is used to check if a point P is inside the triangle P1P2P3 or not	
intextnodes.m	23
main.m	
Fafdfdfsdffsdfs	
$ I_2 = \left \int_0^T \psi(t) \left\{ u(a,t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta,t)} \int_a^\theta c(\xi) u_t(\xi,t) d\xi \right\} dt \right $	
	24
malla01.m	25
provideGeometricData.m	25
refineRGB.m	
Cprintf([1 0 0],mami estoy vivo)	
shownu.m	26

Chapter 6

Class Documentation

6.1 cell Class Reference

A MatLab cell array or matrix.

6.1.1 Detailed Description

A MatLab cell array or matrix.

This class is an artificially created class in doxygen to allow more precise type declarations

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

- [class_substitutes.c](#)

6.2 char Class Reference

A MatLab character array.

6.2.1 Detailed Description

A MatLab character array.

This class is an artificially created class in doxygen to allow more precise type declarations and represents string-like types.

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

- [class_substitutes.c](#)

6.3 colvec Class Reference

A matlab column vector.

6.3.1 Detailed Description

A matlab column vector.

This class is an artificially created class in doxygen to allow more precise type declarations

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

- [class_substitutes.c](#)

6.4 double Class Reference

A double value.

6.4.1 Detailed Description

A double value.

This class is an artificially created class in doxygen to allow more precise type declarations. The MatLab type associated with this class is double.

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

- [class_substitutes.c](#)

6.5 function_handle Class Reference

A MatLab function handle.

6.5.1 Detailed Description

A MatLab function handle.

This class is an artificially created class in doxygen to allow more precise type declarations

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

- [class_substitutes.c](#)

6.6 handle Class Reference

Matlab's base handle class (documentation generation substitute)

Public Attributes

- matlabtypesubstitute [addlistener](#)
Creates a listener for the specified event and assigns a callback function to execute when the event occurs.
- matlabtypesubstitute [notify](#)
Broadcast a notice that a specific event is occurring on a specified handle object or array of handle objects.
- matlabtypesubstitute [delete](#)
Handle object destructor method that is called when the object's lifecycle ends.
- matlabtypesubstitute [disp](#)
Handle object disp method which is called by the display method. See the MATLAB disp function.
- matlabtypesubstitute [display](#)
Handle object display method called when MATLAB software interprets an expression returning a handle object that is not terminated by a semicolon. See the MATLAB display function.
- matlabtypesubstitute [findobj](#)
Finds objects matching the specified conditions from the input array of handle objects.
- matlabtypesubstitute [findprop](#)
Returns a meta.property objects associated with the specified property name.
- matlabtypesubstitute [fields](#)
Returns a cell array of string containing the names of public properties.
- matlabtypesubstitute [fieldnames](#)
Returns a cell array of string containing the names of public properties. See the MATLAB fieldnames function.
- matlabtypesubstitute [isvalid](#)
Returns a logical array in which elements are true if the corresponding elements in the input array are valid handles. This method is Sealed so you cannot override it in a handle subclass.
- matlabtypesubstitute [eq](#)
Relational functions example. See details for more information.
- matlabtypesubstitute [transpose](#)
Transposes the elements of the handle object array.
- matlabtypesubstitute [permute](#)
Rearranges the dimensions of the handle object array. See the MATLAB permute function.
- matlabtypesubstitute [reshape](#)
changes the dimensions of the handle object array to the specified dimensions. See the MATLAB reshape function.
- matlabtypesubstitute [sort](#)
ort the handle objects in any array in ascending or descending order.

6.6.1 Detailed Description

Matlab's base handle class (documentation generation substitute)

As doxygen does not know the class "handle" from itself, many classes do not get rendered within the doxygen output. This workaround guarantees a correct (also graphical) representation of the class hierarchy.

Note here that by having the type handle it could also mean to have a vector or matrix of handles.

6.6.2 Member Data Documentation

6.6.2.1 matlabtypesubstitute `handle::addlistener`

Creates a listener for the specified event and assigns a callback function to execute when the event occurs.

See also

[notify](#)

6.6.2.2 matlabtypesubstitute `handle::delete`

Handle object destructor method that is called when the object's lifecycle ends.

6.6.2.3 matlabtypesubstitute `handle::disp`

Handle object disp method which is called by the display method. See the MATLAB disp function.

6.6.2.4 matlabtypesubstitute `handle::display`

Handle object display method called when MATLAB software interprets an expression returning a handle object that is not terminated by a semicolon. See the MATLAB display function.

6.6.2.5 matlabtypesubstitute `handle::eq`

Relational functions example. See details for more information.

Other possible relational operators:

`-ne -lt -le -gt -ge`

Relational functions return a logical array of the same size as the pair of input handle object arrays. Comparisons use a number associated with each handle. You can assume that the same two handles will compare as equal and the repeated comparison of any two handles will yield the same result in the same MATLAB session. Different handles are always not-equal. The order of handles is purely arbitrary, but consistent.

6.6.2.6 matlabtypesubstitute `handle::fieldnames`

Returns a cell array of string containing the names of public properties. See the MATLAB fieldnames function.

6.6.2.7 matlabtypesubstitute handle::fields

Returns a cell array of string containing the names of public properties.

6.6.2.8 matlabtypesubstitute handle::findobj

Finds objects matching the specified conditions from the input array of handle objects.

6.6.2.9 matlabtypesubstitute handle::findprop

Returns a meta.property objects associated with the specified property name.

6.6.2.10 matlabtypesubstitute handle::isvalid

Returns a logical array in which elements are true if the corresponding elements in the input array are valid handles. This method is Sealed so you cannot override it in a handle subclass.

6.6.2.11 matlabtypesubstitute handle::notify

Broadcast a notice that a specific event is occurring on a specified handle object or array of handle objects.

6.6.2.12 matlabtypesubstitute handle::permute

Rearranges the dimensions of the handle object array. See the MATLAB permute function.

6.6.2.13 matlabtypesubstitute handle::reshape

Changes the dimensions of the handle object array to the specified dimensions. See the MATLAB reshape function.

6.6.2.14 matlabtypesubstitute handle::sort

Sorts the handle objects in any array in ascending or descending order.

The order of handles is purely arbitrary, but reproducible in a given MATLAB session. See the MATLAB sort function.

6.6.2.15 matlabtypesubstitute handle::transpose

Transposes the elements of the handle object array.

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

- [class_substitutes.c](#)

6.7 integer Class Reference

An integer value.

6.7.1 Detailed Description

An integer value.

This class is an artificially created class in doxygen to allow more precise type declarations. Matlab types associated with this class are all int-types (int8, uint8 etc).

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

- [class_substitutes.c](#)

6.8 logical Class Reference

A boolean value.

6.8.1 Detailed Description

A boolean value.

This class can be seen as synonym for boolean values/flags used inside classes. In order to stick with matlab conventions/datatypes, this class was named logical instead of bool or boolean.

This class is an artificially created class in doxygen to allow more precise type declarations

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

- [class_substitutes.c](#)

6.9 rowvec Class Reference

A matlab row vector.

6.9.1 Detailed Description

A matlab row vector.

This class is an artificially created class in doxygen to allow more precise type declarations

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

- [class_substitutes.c](#)

6.10 struct Class Reference

A MatLab struct.

6.10.1 Detailed Description

A MatLab struct.

This class is an artificially created class in doxygen to allow more precise type declarations

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

- [class_substitutes.c](#)

6.11 varargin Class Reference

A variable number of input arguments.

6.11.1 Detailed Description

A variable number of input arguments.

This class is an artificially created class in doxygen to allow more precise type declarations.

For more information about the varargin argument see the [MatLab documentation on varargin](#).

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

- [class_substitutes.c](#)

6.12 varargout Class Reference

A variable number of output arguments.

6.12.1 Detailed Description

A variable number of output arguments.

This class is an artificially created class in doxygen to allow more precise type declarations.

For more information about the varargout argument see the [MatLab documentation on varargout](#).

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

- [class_substitutes.c](#)

Chapter 7

File Documentation

7.1 auxiliar.m File Reference

Functions

- noret::substitute [genbasftn](#) (matlabtypesubstitute u, matlabtypesubstitute i, matlabtypesubstitute jk)

7.1.1 Function Documentation

7.1.1.1 noret::substitute [genbasftn](#) (matlabtypesubstitute *u*, matlabtypesubstitute *i*, matlabtypesubstitute *jk*)

7.2 Bases_func_Maxwell.m File Reference

Calcula el valor en el punto x de la función base ϕ_j^i (elementos de Whitney 2D) definido por

$$\begin{aligned}\phi_1^i &= \lambda_1 \cdot \nabla \lambda_2 - \lambda_2 \cdot \nabla \lambda_1 \\ \phi_2^i &= \lambda_1 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_1 \\ \phi_3^i &= \lambda_2 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_2\end{aligned}$$

Functions

- mlhsInnerSubst
< matlabtypesubstitute, v > [Bases_func_Maxwell](#) (matlabtypesubstitute u, matlabtypesubstitute i, matlabtypesubstitute j, matlabtypesubstitute x)

Calcula el valor en el punto x de la función base ϕ_j^i (elementos de Whitney 2D) definido por

$$\begin{aligned}\phi_1^i &= \lambda_1 \cdot \nabla \lambda_2 - \lambda_2 \cdot \nabla \lambda_1 \\ \phi_2^i &= \lambda_1 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_1 \\ \phi_3^i &= \lambda_2 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_2\end{aligned}$$

7.2.1 Detailed Description

Calcula el valor en el punto x de la función base ϕ_j^i (elementos de Whitney 2D) definido por

$$\begin{aligned}\phi_1^i &= \lambda_1 \cdot \nabla \lambda_2 - \lambda_2 \cdot \nabla \lambda_1 \\ \phi_2^i &= \lambda_1 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_1 \\ \phi_3^i &= \lambda_2 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_2\end{aligned}$$

7.2.2 Function Documentation

7.2.2.1 `mlhsInnerSubst< matlabtypesubstitute, v > Bases_func_Maxwell(matlabtypesubstitute u , matlabtypesubstitute i , matlabtypesubstitute j , matlabtypesubstitute x)`

Calcula el valor en el punto x de la función base ϕ_j^i (elementos de Whitney 2D) definido por

$$\begin{aligned}\phi_1^i &= \lambda_1 \cdot \nabla \lambda_2 - \lambda_2 \cdot \nabla \lambda_1 \\ \phi_2^i &= \lambda_1 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_1 \\ \phi_3^i &= \lambda_2 \cdot \nabla \lambda_3 - \lambda_3 \cdot \nabla \lambda_2\end{aligned}$$

Author

Ricardo Prato

Date

2012-03-20

Change in 1.1 (Catalina Dominguez, 2012-04-16)

Parameters

u	Estructura
i	Id del elemento en la triangulación
j	Id de la función base
x	punto

Return values

v	$\phi_j^i(x)$
-----	---------------

Required fields of u:

- `u.alfa` -- coordenadas baricentricas asociadas al elemento i

7.3 carst_basftn.m File Reference

Functions

- `noret::substitute carst_basftn` (matlabtypesubstitute u, matlabtypesubstitute ik, matlabtypesubstitute jk)
- `mlhsInnerSubst`
`< matlabtypesubstitute, True > mtoc_subst_carst_basftn_m_tsbust_cotm_test_line` (matlabtypesubstitute P, matlabtypesubstitute P1, matlabtypesubstitute P2)

7.3.1 Function Documentation

7.3.1.1 `noret::substitute carst_basftn (matlabtypesubstitute u, matlabtypesubstitute ik, matlabtypesubstitute jk)`

7.3.1.2 `mlhsInnerSubst<matlabtypesubstitute,True> mtoc_subst_carst_basftn_m_tsbust_cotm_test_line (matlabtypesubstitute P, matlabtypesubstitute P1, matlabtypesubstitute P2)`

7.4 class_substitutes.c File Reference

Classes

- class `matrix`
A matlab matrix.
- class `sparsematrix`
A matlab sparse matrix.
- class `handle`
Matlab's base handle class (documentation generation substitute)

7.5 developers.c File Reference

7.6 funciones.m File Reference

Functions

- mlhsInnerSubst
< matlabtypesubstitute, value > [funciones](#) (matlabtypesubstitute rhs)

7.6.1 Function Documentation

7.6.1.1 mlhsInnerSubst<matlabtypesubstitute,value> [funciones](#) (matlabtypesubstitute *rhs*)

7.7 genbasftn.m File Reference

Functions

- noret::substitute [genbasftn](#) (matlabtypesubstitute u, matlabtypesubstitute i, matlabtypesubstitute jk)

7.7.1 Function Documentation

7.7.1.1 noret::substitute [genbasftn](#) (matlabtypesubstitute *u*, matlabtypesubstitute *i*, matlabtypesubstitute *jk*)

7.8 inside_triangle.m File Reference

inside_triangle is used to check if a point P is inside the triangle P1P2P3 or not.

Functions

- mlhsInnerSubst
< matlabtypesubstitute, True > [inside_triangle](#) (matlabtypesubstitute P, matlabtypesubstitute P1, matlabtypesubstitute P2, matlabtypesubstitute P3)
inside_triangle is used to check if a point P is inside the triangle P1P2P3 or not.

7.8.1 Detailed Description

inside_triangle is used to check if a point P is inside the triangle P1P2P3 or not.

7.8.2 Function Documentation

7.8.2.1 mlhsInnerSubst< matlabtypesubstitute, True > inside_triangle (matlabtypesubstitute *P*, matlabtypesubstitute *P1*, matlabtypesubstitute *P2*, matlabtypesubstitute *P3*)

inside_triangle is used to check if a point *P* is inside the triangle *P1P2P3* or not.

Inputs: *P*, *P1*, *P2* and *P3* are vectors of length 2 or three of the form [x y z] or [x y]

Output: True True=1 => *P* is on or inside *P1P2P3* True=0 => *P* is outside *P1P2P3*

Example

```
True=inside_triangle([0.5 0.5],[0 0],[0 2],[2 0]);
```

The following algorithm is implemented If *P* is ON or INSIDE the triangle

```
Area(PP1P2) + Area(PP2P3) + Area(PP3P1) = Area(P1P2P3)
```

If *P* is OUTSIDE then,

```
Area(PP1P2) + Area(PP2P3) + Area(PP3P1) > Area(P1P2P3)
```

Area of a triangle can be found using the determinant

$$\text{Area} = \text{abs}(1/2 * \begin{vmatrix} x1 & y1 & 1 \\ x2 & y2 & 1 \\ x3 & y3 & 1 \end{vmatrix})$$

Parameters

<i>P</i>	<i>P</i>
<i>P1</i>	<i>P1</i>
<i>P2</i>	<i>P2</i>
<i>P3</i>	<i>P3</i>

Return values

<i>True</i>	True
-------------	------

7.9 intextnodes.m File Reference

Functions

- mlhsInnerSubst
< matlabtypesubstitute, u > [intextnodes](#) (matlabtypesubstitute u)
- mlhsInnerSubst
< matlabtypesubstitute, alfa > [mtoc_subst_intextnodes_m_tsbu cotm_invM](#) (matlabtypesubstitute p, matlabtypesubstitute q, matlabtypesubstitute r)

7.9.1 Function Documentation

7.9.1.1 mlhsInnerSubst<matlabtypesubstitute,u> intextnodes (matlabtypesubstitute *u*)

7.9.1.2 mlhsInnerSubst<matlabtypesubstitute,alfa> mtoc_subst_intextnodes_m_tsbu s_cotm_invM (matlabtypesubstitute *p*, matlabtypesubstitute *q*, matlabtypesubstitute *r*)

7.10 main.m File Reference

fafdfdfsfdfsfds

$$|I_2| = \left| \int_0^T \psi(t) \left\{ u(a,t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta,t)} \int_a^\theta c(\xi) u_t(\xi,t) d\xi \right\} dt \right|$$

Functions

- mlhsInnerSubst
< matlabtypesubstitute, u > **main** (matlabtypesubstitute *iter*, matlabtypesubstitute *rhs*, matlabtypesubstitute *mode*)

fafdfdfsfdfsfds

$$|I_2| = \left| \int_0^T \psi(t) \left\{ u(a,t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta,t)} \int_a^\theta c(\xi) u_t(\xi,t) d\xi \right\} dt \right|$$

7.10.1 Detailed Description

fafdfdfsfdfsfds

$$|I_2| = \left| \int_0^T \psi(t) \left\{ u(a,t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta,t)} \int_a^\theta c(\xi) u_t(\xi,t) d\xi \right\} dt \right|$$

7.10.2 Function Documentation

7.10.2.1 mlhsInnerSubst< matlabtypesubstitute, u > **main** (matlabtypesubstitute *iter*, matlabtypesubstitute *rhs*, matlabtypesubstitute *mode*)

fafdfdfsfdfsfds

$$|I_2| = \left| \int_0^T \psi(t) \left\{ u(a,t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta,t)} \int_a^\theta c(\xi) u_t(\xi,t) d\xi \right\} dt \right|$$

Parameters

<i>iter</i>	iter
<i>rhs</i>	rhs
<i>mode</i>	mode

Return values

u	u
-----	-----

7.11 malla01.m File Reference

Functions

- `mlhsInnerSubst`
`< matlabtypesubstitute, u > malla01 (matlabtypesubstitute mode)`

7.11.1 Function Documentation

7.11.1.1 `mlhsInnerSubst<matlabtypesubstitute,u> malla01 (matlabtypesubstitute mode)`

7.12 provideGeometricData.m File Reference

Functions

- `mlhsSubst< mlhsInnerSubst`
`< matlabtypesubstitute,`
`edge2nodes >,mlhsInnerSubst`
`< matlabtypesubstitute,`
`element2edges >,mlhsInnerSubst`
`< matlabtypesubstitute,`
`varargout > > provideGeometricData (matlabtypesubstitute elements, matlabtypesubstitute varargin)`

7.12.1 Function Documentation

7.12.1.1 `mlhsSubst<mlhsInnerSubst<matlabtypesubstitute,edge2nodes> ,mlhsInner-`
`Subst<matlabtypesubstitute,element2edges> ,mlhsInnerSubst<matlabtypesubstitute,varargout>`
`> provideGeometricData (matlabtypesubstitute elements, matlabtypesubstitute varargin)`

7.13 refineRGB.m File Reference

```
cprintf([1 0 0],mami estoy vivo)
```

Functions

- mlhsSubst< mlhsInnerSubst
 < matlabtypesubstitute,
 coordinates >,mlhsInnerSubst
 < matlabtypesubstitute,
 newElements >,mlhsInnerSubst
 < matlabtypesubstitute,
[varargout](#) > > [refineRGB](#) (matlabtypesubstitute coordinates, matlabtypesubstitute elements, matlabtypesubstitute [varargin](#))
`cprintf([1 0 0],mami estoy vivo)`

7.13.1 Detailed Description

`cprintf([1 0 0],mami estoy vivo)`

7.13.2 Function Documentation

7.13.2.1 mlhsSubst< mlhsInnerSubst< matlabtypesubstitute, coordinates >,mlhsInnerSubst< matlabtypesubstitute, newElements >,mlhsInnerSubst< matlabtypesubstitute, [varargout](#) > > [refineRGB](#) (matlabtypesubstitute *coordinates*, matlabtypesubstitute *elements*, matlabtypesubstitute *varargin*)

`cprintf([1 0 0],mami estoy vivo)`

Parameters

<i>coordinates</i>	coordinates
<i>elements</i>	elements
<i>varargin</i>	varargin

Return values

<i>coordinates</i>	coordinates
<i>newElements</i>	newElements
<i>varargout</i>	varargout

7.14 shownu.m File Reference

Functions

- noret::substitute [shownu](#) (matlabtypesubstitute elements, matlabtypesubstitute nodes)

7.14.1 Function Documentation

7.14.1.1 `noret::substitute shownu (matlabtypesubstitute elements, matlabtypesubstitute nodes)`