

Resolucion de sistemas lineales

Generated by Doxygen 1.8.8

Tue Mar 14 2017 19:09:27

Contents

Chapter 1

Todo List

Class [MatrizBand](#)

- Hacer comportamiento para cambiar tama banda
- Multiplicaci matrices

Class [MatrizBandComp](#)

- Multiplicaci matrices

Class [MatrizDispersa](#)

- Hacer comportamiento para cambiar tama banda
- Multiplicaci matrices

Class [ResuelveCGMBandDisp](#)

- Definir clase de producto interior y poder pasarlo como argumento y user este

Chapter 2

Bug List

| | |
|--|--------------------------|
| Class BCGM | No hay errores conocidos |
| Class Ctrl_visualizacion | No hay errores conocidos |
| Class Matriz | No hay errores conocidos |
| Class Matriz_Base | No hay errores conocidos |
| Class MatrizBand | No hay errores conocidos |
| Class MatrizBandComp | No hay errores conocidos |
| Class MatrizBandDisp | No hay errores conocidos |
| Class MatrizDensa | No hay errores conocidos |
| Class MatrizDispersa | No hay errores conocidos |
| Class MultOp | No hay errores conocidos |
| Class ProductoPunto | No hay errores conocidos |
| Class ResuelveCGM | No hay errores conocidos |
| Class ResuelveCGMBandDisp | No hay errores conocidos |
| Class ResuelveFactorizacionCholeski | No hay errores conocidos |
| Class ResuelveFactorizacionCholeskiBandDisp | No hay errores conocidos |
| Class ResuelveFactorizacionLU | No hay errores conocidos |
| Class ResuelveFactorizacionLUBandDisp | No hay errores conocidos |
| Class ResuelveGaussSeidel | No hay errores conocidos |
| Class ResuelveGaussSeidelBandDisp | No hay errores conocidos |
| Class ResuelveInversa | No hay errores conocidos |
| Class ResuelveJacobi | No hay errores conocidos |

- Class [ResuelveJacobiBandDisp](#)**
 - No hay errores conocidos
- Class [ResuelveSistemaLineal](#)**
 - No hay errores conocidos
- Class [ResuelveTridiagonal](#)**
 - No hay errores conocidos
- Class [Vector](#)**
 - No hay errores conocidos
- Class [Vector_Base](#)**
 - No hay errores conocidos

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

| | |
|---|----|
| Ctrl_visualizacion | ?? |
| Matriz_Base | ?? |
| Matriz | ?? |
| MatrizBandDisp | ?? |
| MatrizBand | ?? |
| MatrizBandComp | ?? |
| MatrizDispersa | ?? |
| MatrizDensa | ?? |
| MultOp | ?? |
| ResuelveCGM | ?? |
| ResuelveCGMBandDisp | ?? |
| ProductoPunto | ?? |
| ResuelveCGM | ?? |
| ResuelveSistemaLineal | ?? |
| BCGM | ?? |
| ResuelveCGM | ?? |
| ResuelveFactorizacionCholeski | ?? |
| ResuelveFactorizacionCholeskiBandDisp | ?? |
| ResuelveFactorizacionLU | ?? |
| ResuelveFactorizacionLUBandDisp | ?? |
| ResuelveGaussSeidel | ?? |
| ResuelveGaussSeidelBandDisp | ?? |
| ResuelveInversa | ?? |
| ResuelveJacobi | ?? |
| ResuelveJacobiBandDisp | ?? |
| ResuelveTridiagonal | ?? |
| Vector_Base | ?? |
| Vector | ?? |

Chapter 4

Class Index

4.1 Class List

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

| | | |
|---|--|----|
| BCGM | Clase para resolucil sistema lineal mediante CGM | ?? |
| Ctrl_visualizacion | Clase para visualizar nmeros sobrecargando la funcion | ?? |
| Matriz | Clase base para el trabajar con matrices de punto flotante | ?? |
| Matriz_Base | Clase base para el trabajar con matrices | ?? |
| MatrizBand | | ?? |
| MatrizBandComp | | ?? |
| MatrizBandDisp | Clase base para trabajar con matrices bandadas y dispersas | ?? |
| MatrizDensa | Clase para el trabajar con matrices densas de punto flotante | ?? |
| MatrizDispersa | | ?? |
| MultOp | Clase para implementar la multiplicaci la matriz por el vector | ?? |
| ProductoPunto | Clase para implementar el producto punto | ?? |
| ResuelveCGM | Clase para resolucil sistema lineal mediante CGM standard | ?? |
| ResuelveCGMBandDisp | Clase para resolucil sistema lineal mediante CGM | ?? |
| ResuelveFactorizacionCholeski | Clase para resolucil sistema lineal usando factorizacion Choleski | ?? |
| ResuelveFactorizacionCholeskiBandDisp | Clase para resolucil sistema lineal usando factorizacion Choleski | ?? |
| ResuelveFactorizacionLU | Clase para resolucil sistema lineal usando factorizacion LU | ?? |
| ResuelveFactorizacionLUBandDisp | Clase para resolucil sistema lineal usando factorizacion LU | ?? |
| ResuelveGaussSeidel | Clase para resolucil sistema lineal mediante Gauss-Seidel | ?? |
| ResuelveGaussSeidelBandDisp | Clase para resolucil sistema lineal mediante Gauss-Seidel | ?? |
| ResuelveInversa | Clase para resolucil sistema lineal mediante el uso de la matriz inversa | ?? |
| ResuelveJacobi | Clase para resolucil sistema lineal mediante Jacobi | ?? |
| ResuelveJacobiBandDisp | Clase para resolucil sistema lineal mediante Jacobi | ?? |
| ResuelveSistemaLineal | Clase para resolucil sistema lineal | ?? |
| ResuelveTridiagonal | Clase para resolucil sistema lineal tridiagonal | ?? |
| Vector | Clase para el trabajar con vectores de punto flotante | ?? |
| Vector_Base | Clase base para el trabajar con vectores | ?? |

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

| | |
|---|----|
| BCGM.cpp | ?? |
| BCGM.hpp | ?? |
| Ctrl_vis.hpp | ?? |
| Definiciones.hpp | ?? |
| Ejemplo.cpp | ?? |
| Matriz.cpp | ?? |
| Matriz.hpp | ?? |
| Matriz_Base.cpp | ?? |
| Matriz_Base.hpp | ?? |
| MatrizBand.cpp | ?? |
| MatrizBand.hpp | ?? |
| MatrizBandComp.cpp | ?? |
| MatrizBandComp.hpp | ?? |
| MatrizBandDisp.hpp | ?? |
| MatrizDensa.cpp | ?? |
| MatrizDensa.hpp | ?? |
| MatrizDispersa.cpp | ?? |
| MatrizDispersa.hpp | ?? |
| MultOp.hpp | ?? |
| ProductoPunto.hpp | ?? |
| ResuelveCGM.hpp | ?? |
| ResuelveCGMBandDisp.hpp | ?? |
| ResuelveFactorizacionCholeski.hpp | ?? |
| ResuelveFactorizacionCholeskiBandDisp.cpp | ?? |
| ResuelveFactorizacionCholeskiBandDisp.hpp | ?? |
| ResuelveFactorizacionLU.hpp | ?? |
| ResuelveFactorizacionLUBandDisp.cpp | ?? |
| ResuelveFactorizacionLUBandDisp.hpp | ?? |
| ResuelveGaussSeidel.hpp | ?? |
| ResuelveGaussSeidelBandDisp.cpp | ?? |
| ResuelveGaussSeidelBandDisp.hpp | ?? |
| ResuelveInversa.cpp | ?? |
| ResuelveInversa.hpp | ?? |
| ResuelveJacobi.hpp | ?? |
| ResuelveJacobiBandDisp.cpp | ?? |
| ResuelveJacobiBandDisp.hpp | ?? |
| ResuelveSistemaLineal.hpp | ?? |
| ResuelveTridiagonal.cpp | ?? |
| ResuelveTridiagonal.hpp | ?? |
| Vector.cpp | ?? |
| Vector.hpp | ?? |
| Vector_Base.hpp | ?? |

Chapter 6

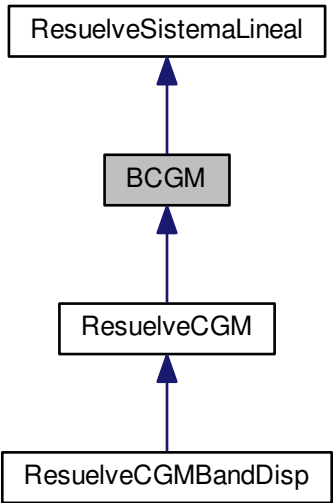
Class Documentation

6.1 BCGM Class Reference

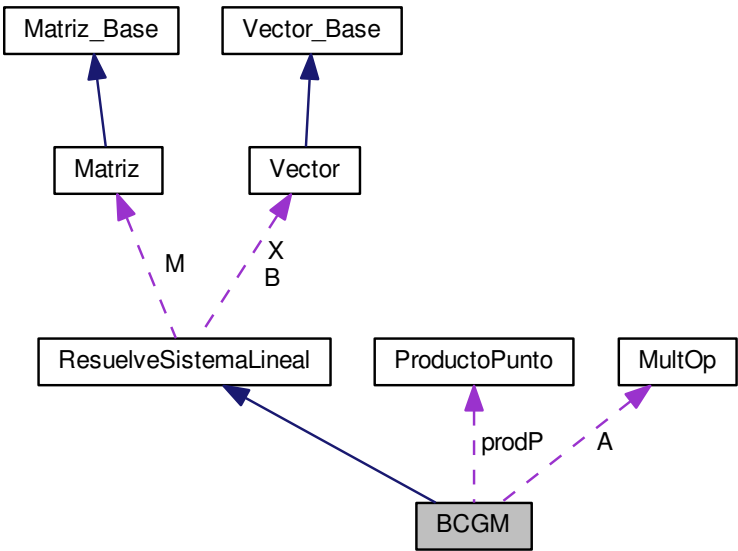
Clase para resolucil sistema lineal mediante CGM.

```
#include <BCGM.hpp>
```

Inheritance diagram for BCGM:



Collaboration diagram for BCGM:



Public Member Functions

- **BCGM** (**MultOp** &a, **ProductoPunto** &prod, int iter, **ldouble** ep)
Constructor de la clase.
- void **resuelve** (void)
Resuelve el sistema lineal.
- void **resuelve** (**Vector** *x, **Vector** *b)
Resuelve el sistema lineal.
- void **tolerancia** (**ldouble** eps)
Establece la tolerancia del metodo.
- void **iteraciones** (int iter)
Establece el maximo numero de iteraciones.

Protected Attributes

- **MultOp** * **A**
- **ProductoPunto** * **prodP**
Producto Punto.
- int **Iter**
Nmero mmo de iteraciones.
- **ldouble** **Ep**
Tolerancia.

Additional Inherited Members

6.1.1 Detailed Description

Clase para resolucil sistema lineal mediante CGM.

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.1.2 Constructor & Destructor Documentation

6.1.2.1 `BCGM::BCGM (MultOp & a, ProductoPunto & prod, int iter, Idouble ep)` `[inline]`

Constructor de la clase.

Parameters

| | |
|-------------|---|
| <i>a</i> | Puntero a la implementaci la matriz por el vector |
| <i>prod</i> | Puntero a la implementacil producto punto de dos vectores |
| <i>iter</i> | Mmo nmero de iteraciones |
| <i>ep</i> | Tolerancia mma |

6.1.3 Member Function Documentation

6.1.3.1 `void BCGM::iteraciones (int iter)` `[inline]`

Establece el maximo numero de iteraciones.

Parameters

| | |
|-------------|------------------------|
| <i>iter</i> | Iteraciones del metodo |
|-------------|------------------------|

6.1.3.2 `void BCGM::resuelve (void)` `[virtual]`

Resuelve el sistema lineal.

Implements [ResuelveSistemaLineal](#).

6.1.3.3 `void BCGM::resuelve (Vector * x, Vector * b)` `[inline],[virtual]`

Resuelve el sistema lineal.

Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un vector, solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

Implements [ResuelveSistemaLineal](#).

6.1.3.4 `void BCGM::tolerancia (Idouble eps)` `[inline]`

Establece la tolerancia del metodo.

Parameters

| | |
|------------|-----------------------|
| <i>eps</i> | Tolerancia del metodo |
|------------|-----------------------|

6.1.4 Member Data Documentation

6.1.4.1 `MultOp* BCGM::A` `[protected]`

6.1.4.2 `Idouble BCGM::Ep` `[protected]`

Tolerancia.

6.1.4.3 `int BCGM::Iter` `[protected]`

Nmero mmo de iteraciones.

6.1.4.4 `ProductoPunto* BCGM::prodP` `[protected]`

Producto Punto.

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

- [BCGM.hpp](#)
- [BCGM.cpp](#)

6.2 Ctrl_visualizacion Class Reference

Clase para visualizar nmeros sobrecargando la funcion.

```
#include <Ctrl_vis.hpp>
```

Public Member Functions

- void [visualiza](#) (char v)
Visualiza un nmero char.
- void [visualiza](#) (int v)
Visualiza un nmero int.
- void [visualiza_e](#) (double v)
Visualiza un nmero double con formato %+1.10le.
- void [visualiza_en](#) (double v)
Visualiza un nmero double con formato %+1.10le y cambia de linea.
- void [visualiza_n](#) (double v)
Visualiza un nmero double con formato %+1.10lf.
- void [visualiza_nn](#) (double v)
Visualiza un nmero double con formato %+1.10lf y cambia de linea.
- void [visualiza_e](#) (long double v)
Visualiza un nmero long double con formato %+1.10Le.
- void [visualiza_en](#) (long double v)
Visualiza un nmero long double con formato %+1.10Le y cambia de linea.
- void [visualiza_n](#) (long double v)
Visualiza un nmero long double con formato %+1.10Lf.
- void [visualiza_nn](#) (long double v)
Visualiza un nmero long double con formato %+1.10Lf y cambia de linea.

6.2.1 Detailed Description

Clase para visualizar nmeros sobrecargando la funcion.

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.2.2 Member Function Documentation

6.2.2.1 void Ctrl_visualizacion::visualiza (char v) [inline]

Visualiza un nmero char.

6.2.2.2 void Ctrl_visualizacion::visualiza (int v) [inline]

Visualiza un nmero int.

6.2.2.3 void Ctrl_visualizacion::visualiza_e (double v) [inline]

Visualiza un nmero double con formato %+1.10le.

6.2.2.4 void Ctrl_visualizacion::visualiza_e (long double v) [inline]

Visualiza un nmero long double con formato %+1.10Le.

6.2.2.5 `void Ctrl_visualizacion::visualiza_en (double v) [inline]`

Visualiza un nmero double con formato `%+1.10le` y cambia de linea.

6.2.2.6 `void Ctrl_visualizacion::visualiza_en (long double v) [inline]`

Visualiza un nmero long double con formato `%+1.10Le` y cambia de linea.

6.2.2.7 `void Ctrl_visualizacion::visualiza_n (double v) [inline]`

Visualiza un nmero double con formato `%+1.10lf`.

6.2.2.8 `void Ctrl_visualizacion::visualiza_n (long double v) [inline]`

Visualiza un nmero long double con formato `%+1.10Lf`.

6.2.2.9 `void Ctrl_visualizacion::visualiza_nn (double v) [inline]`

Visualiza un nmero double con formato `%+1.10lf` y cambia de linea.

6.2.2.10 `void Ctrl_visualizacion::visualiza_nn (long double v) [inline]`

Visualiza un nmero long double con formato `%+1.10Lf` y cambia de linea.

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

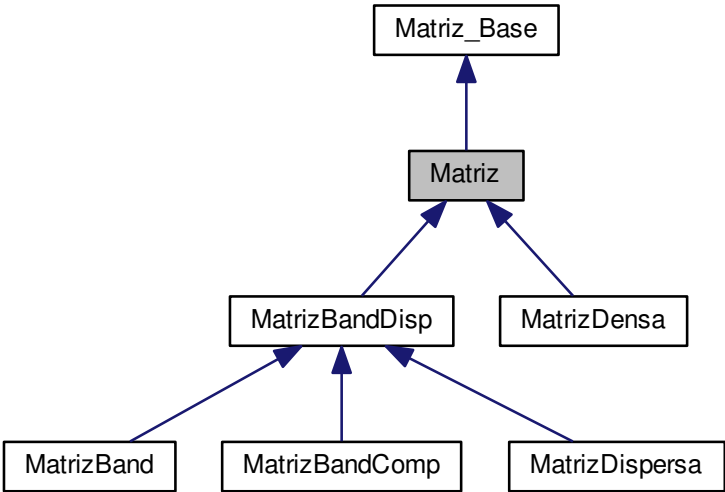
- [Ctrl_vis.hpp](#)

6.3 Matriz Class Reference

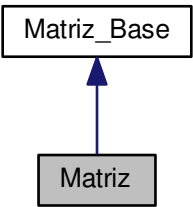
Clase base para el trabajar con matrices de punto flotante.

```
#include <Matriz.hpp>
```

Inheritance diagram for Matriz:



Collaboration diagram for Matriz:



Public Member Functions

- [Matriz](#) (void)
Constructor de la clase.
- [ldouble operator\(\)](#) (size_t ren, size_t col)
Retorna el valor del renglon y columna solicitado.
- virtual void [asigna](#) (const int ren, const int col, const [ldouble](#) val)=0
Asigna el valor indicado en el renglon y columna solicitado.
- virtual [ldouble retorna](#) (const int ren, const int col)=0
Retorna el valor del renglon y columna solicitado.
- void [retornaRenglon](#) (int r, [ldouble](#) *arr, int tam)
Retorna el contenido del renglon de la matriz en el arreglo.
- virtual void [inicializa](#) (const [ldouble](#) val)=0
Inicializa la matriz con el valor indicado.
- void [inicializaDiagonal](#) (const [ldouble](#) val)
Inicializa la matriz como una matriz diagonal con el valor indicado.
- void [visualiza](#) (const int tp)
Visualiza la matriz.
- void [convierte](#) ([ldouble](#) *a, int ren, int tam)
Convierte el arreglo unidimensional en parte de la matriz en el renglon Ren y de tamam.
- void [convierte](#) ([ldouble](#) **a, int ren, int col)
Convierte un arreglo bidimensional de numero de renglones REN y columnas Col.
- void [convierte](#) (int ren, [Vector](#) *a)
Convierte el contenido del vector en parte de la matriz en el renglon Ren.
- void [convierte](#) (int ren, int col, [ldouble](#) *a)
Convierte el contenido del arreglo unidimensional que contiene una matriz de Ren x Col a matriz.
- void [retornaVector](#) ([Vector](#) *a)
Retorna todo el contenido de la matriz como un vector.
- virtual void [copia](#) ([Matriz](#) *a)
Copia el contenido de la matriz a la matriz pasada como partro.
- void [suma](#) ([Matriz](#) *a, [Matriz](#) *b)
Suma las matrices A y B.
- void [suma](#) ([Matriz](#) *a)
Suma a la matriz el contenido de la matriz pasada como partro.
- void [resta](#) ([Matriz](#) *a, [Matriz](#) *b)
Resta a la matriz A la matriz B.
- void [resta](#) ([Matriz](#) *a)
Resta a la matriz el contenido de la matriz padada como partro.
- virtual void [multiplica](#) ([ldouble](#) esc)=0
Multiplica la matriz por el escalar pasado como parametro.
- virtual void [multiplica](#) ([Vector](#) *b, [Vector](#) *r)=0
Multiplica la matriz por el vector B dejando el Resultado en R.
- void [transpuesta](#) ([Matriz](#) *a)
Transpone la matriz pasada como partro.
- int [iguales](#) ([Matriz](#) *a, [ldouble](#) eps=1e-14)
Regresa verdadero si las matrices son iguales, cero en caso contrario.
- int [maximoTamanoColumnaOcupado](#) (void)

- Retorna el mmo tama todas las columnas que sean distintos de cero.*
 - int [entradasDistintasCero](#) (void)
- Retorna el nmero de entradas distintas de cero en la matriz.*
 - virtual int [retornaNumeroColumnasBanda](#) (int ren)=0
- Retorna el nmero de columnas de la banda para el renglon indicado.*
 - virtual int [retornaNumeroColumna](#) (int ren, int ind)=0
- Retorna el nmero de columna cuando se para en el renglon e indice de la banda.*
 - virtual double [retornaValorColumna](#) (int ren, int ind)=0
- Retorna el valor de la columna cuando se para en el renglon e indice de la banda.*

Additional Inherited Members

6.3.1 Detailed Description

Clase base para el trabajar con matrices de punto flotante.

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.3.2 Constructor & Destructor Documentation

6.3.2.1 `Matriz::Matriz (void)` `[inline]`

Constructor de la clase.

6.3.3 Member Function Documentation

6.3.3.1 `virtual void Matriz::asigna (const int ren, const int col, const Idouble val)` `[pure virtual]`

Asigna el valor indicado en el renglon y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |
| <i>val</i> | Valor |

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.3.3.2 `void Matriz::convierte (Idouble * a, int ren, int tam)`

Convierte el arreglo unidimensional en parte de la matriz en el renglon Ren y de tamam.

Parameters

| | |
|------------|--|
| <i>a</i> | puntero al arreglo |
| <i>ren</i> | Renglon en el cual se insertara el contenido del arreglo |
| <i>tam</i> | Tamal arreglo |

6.3.3.3 `void Matriz::convierte (Idouble ** a, int ren, int col)`

Convierte un arreglo bidimensional de numero de renglones REN y columnas Col.

Parameters

| | |
|------------|--------------------|
| <i>a</i> | puntero al arreglo |
| <i>ren</i> | Renglones |
| <i>col</i> | Columnas |

6.3.3.4 void Matriz::convierte (int *ren*, Vector * *a*)

Convierte el contenido del vector en parte de la matriz en el renglon Ren.

Parameters

| | |
|------------|--|
| <i>ren</i> | Renglon en el cual se insertara el contenido del arreglo |
| <i>a</i> | puntero al vector |

6.3.3.5 void Matriz::convierte (int *ren*, int *col*, Idouble * *a*)

Convierte el contenido del arreglo unidimensional que contiene una matriz de Ren x Col a matriz.

Parameters

| | |
|------------|--------------------|
| <i>ren</i> | Nmero de renglones |
| <i>col</i> | Nmero de columnas |
| <i>a</i> | puntero al arreglo |

6.3.3.6 void Matriz::copia (Matriz * *a*) [virtual]

Copia el contenido de la matriz a la matriz pasada como parto.

Parameters

| | |
|----------|--------------------------------------|
| <i>a</i> | Puntero a una Matriz |
|----------|--------------------------------------|

6.3.3.7 int Matriz::entradasDistintasCero (void) [virtual]

Retorna el nmero de entradas distintas de cero en la matriz.

Returns

Nmero de entradas distintas de cero en la matriz

Implements [Matriz_Base](#).

6.3.3.8 int Matriz::iguales (Matriz * *a*, Idouble *eps* = 1e-14)

Regresa verdadero si las matrices son iguales, cero en caso contrario.

Parameters

| | |
|------------|----------------------------------|
| <i>a</i> | Puntero a la matriz |
| <i>eps</i> | Tolerancia usada en la comparaci |

Returns

(1) si son iguales, (0) en caso contrario

6.3.3.9 virtual void Matriz::inicializa (const Idouble *val*) [pure virtual]

Inicializa la matriz con el valor indicado.

Parameters

| | |
|------------|-------|
| <i>val</i> | Valor |
|------------|-------|

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.3.3.10 void Matriz::inicializaDiagonal (const Idouble *val*)

Inicializa la matriz como una matriz diagonal con el valor indicado.

Parameters

| | |
|------------|-------|
| <i>val</i> | Valor |
|------------|-------|

6.3.3.11 `int Matriz::maximoTamanoColumnaOcupado (void)` [virtual]

Retorna el mmo tama todas las columnas que sean distintos de cero.

Returns

Mmo tama columnas ocupadas en la matriz

Implements [Matriz_Base](#).

6.3.3.12 `virtual void Matriz::multiplica (Idouble esc)` [pure virtual]

Multiplica la matriz por el escalar pasado como parametro.

Parameters

| | |
|------------|---------|
| <i>esc</i> | Escalar |
|------------|---------|

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), [MatrizBand](#), and [MatrizBandDisp](#).

6.3.3.13 `virtual void Matriz::multiplica (Vector * b, Vector * r)` [pure virtual]

Multiplica la matriz por el vector B dejando el Resultado en R.

Parameters

| | |
|----------|-------------------------------------|
| <i>b</i> | Puntero a un Vector |
| <i>r</i> | Puntero a un Vector |

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), [MatrizBand](#), and [MatrizBandDisp](#).

6.3.3.14 `Idouble Matriz::operator() (size_t ren, size_t col)` [inline]

Retorna el valor del renglon y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |

Returns

Valor

6.3.3.15 `void Matriz::resta (Matriz * a, Matriz * b)`

Resta a la matriz A la matriz B.

Parameters

| | |
|----------|--------------------------------------|
| <i>a</i> | Puntero a una Matriz |
| <i>b</i> | Puntero a una Matriz |

6.3.3.16 `void Matriz::resta (Matriz * a)`

Resta a la matriz el contenido de la matriz padada como parto.

Parameters

| | |
|----------|--------------------------------------|
| <i>a</i> | Puntero a una Matriz |
|----------|--------------------------------------|

6.3.3.17 `virtual Idouble Matriz::retorna (const int ren, const int col)` [pure virtual]

Retorna el valor del renglon y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |

Returns

Valor

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.3.3.18 `virtual int Matriz::retornaNumeroColumna (int ren, int ind)` `[pure virtual]`

Retorna el nmero de columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Nmero de columna cuando se para en el renglon e indice de la banda

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.3.3.19 `virtual int Matriz::retornaNumeroColumnasBanda (int ren)` `[pure virtual]`

Retorna el nmero de columnas de la banda para el renglon indicado.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
|------------|------------------|

Returns

Nmero de columnas de la banda para el renglon solicitado

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.3.3.20 `void Matriz::retornaRenglon (int r, Idouble * arr, int tam)` `[inline]`

Retorna el contenido del renglon de la matriz en el arreglo.

Parameters

| | | |
|-----|------------|-------------------------|
| | <i>r</i> | Renglon |
| out | <i>arr</i> | Arreglo de tipo Idouble |
| | <i>tam</i> | Tamal arreglo |

6.3.3.21 `virtual double Matriz::retornaValorColumna (int ren, int ind)` `[pure virtual]`

Retorna el valor de la columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Valor de la columna cuando se para en el renglon e indice de la banda

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.3.3.22 `void Matriz::retornaVector (Vector * a)`

Retorna todo el contenido de la matriz como un vector.

Parameters

| | |
|----------|-------------------|
| <i>a</i> | puntero al vector |
|----------|-------------------|

6.3.3.23 void Matriz::suma (Matriz * *a*, Matriz * *b*)

Suma las matrices A y B.

Parameters

| | |
|----------|--------------------------------------|
| <i>a</i> | Puntero a una Matriz |
| <i>b</i> | Puntero a una Matriz |

6.3.3.24 void Matriz::suma (Matriz * *a*)

Suma a la matriz el contenido de la matriz pasada como parto.

Parameters

| | |
|----------|--------------------------------------|
| <i>a</i> | Puntero a una Matriz |
|----------|--------------------------------------|

6.3.3.25 void Matriz::transpuesta (Matriz * *a*)

Transpone la matriz pasada como parto.

Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a la matriz |
|----------|---------------------|

6.3.3.26 void Matriz::visualiza (const int *tp*)

Visualiza la matriz.

Parameters

| | |
|-----------|--|
| <i>tp</i> | (1) Se visualiza el vector de en formato de notacientca, (0) formato notaci punto flotante |
|-----------|--|

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

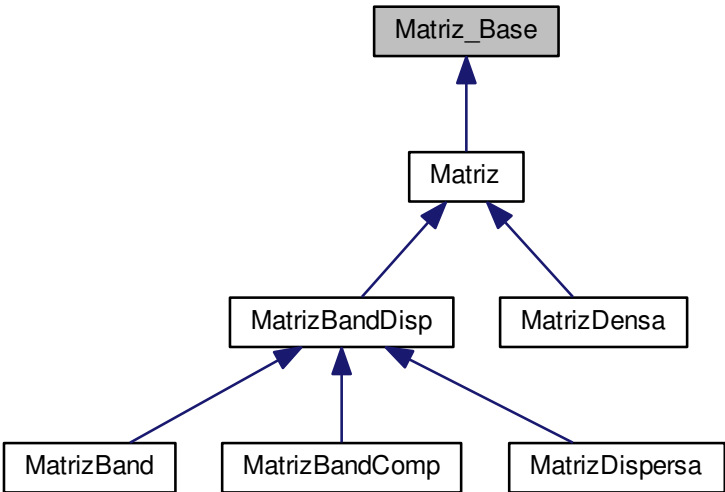
- [Matriz.hpp](#)
- [Matriz.cpp](#)

6.4 Matriz_Base Class Reference

Clase base para el trabajar con matrices.

```
#include <Matriz_Base.hpp>
```

Inheritance diagram for Matriz_Base:



Public Member Functions

- [Matriz_Base](#) (void)
Constructor de la clase.
- [~Matriz_Base](#) ()
Destructor de la clase.
- void [asignaNombre](#) (const char *nmb)
Asigna nombre a la matriz.
- char * [nombre](#) (void)
Retorna el nombre de la matriz.
- int [renglones](#) (void)
Retorna el nmero de renglones de la matriz.
- int [columnas](#) (void)
Retorna el nmero de columnas de la matriz.
- int [matrizCuadrada](#) (void)
Es una matriz cuadrada.
- int [mismaDimension](#) ([Matriz_Base](#) *a)
Revisa si ambas matrices tienen la misma dimensi
int [retornaTipoMatriz](#) (void)
Retorna el tipo de matriz.
void [faltaMemoria](#) (void)
Visualiza el error de falta de memoria para soportar la matriz.
void [visualizaInformacion](#) (void)
Visualiza informacineral de la matriz.
[ldouble tamano](#) (void)
Tama la matriz (aproximado) en Kb.
void [visualizaTamano](#) (void)
Visualiza el tama memoria de la matriz en Kb.
virtual int [maximoTamanoColumnaOcupado](#) (void)=0
Retorna el mmo tama todas las columnas que sean distintos de cero.
virtual int [entradasDistintasCero](#) (void)=0
Retorna el nmero de entradas distintas de cero en la matriz.
virtual void [liberaMemoria](#) (void)=0
Libera la memoria solicitada para la matriz.

Protected Attributes

- int [Col](#)
Nmero de columnas.
- int [Ren](#)
Nmero de renglones.
- int [Ban](#)
Tama la banda (solo si es bandada o dispersa)
- char * [Nmb](#)

- Nombre de la matriz.

– int [Tipo_Matriz](#)

Indica el tipo de matriz de que se trate.

6.4.1 Detailed Description

Clase base para el trabajar con matrices.

Author
Antonio Carrillo Ledesma

Date
primavera 2009

Version
1.0.1

Bug No hay errores conocidos

6.4.2 Constructor & Destructor Documentation

6.4.2.1 `Matriz_Base::Matriz_Base (void)` `[inline]`

Constructor de la clase.

6.4.2.2 `Matriz_Base::~~Matriz_Base ()` `[inline]`

Destructor de la clase.

6.4.3 Member Function Documentation

6.4.3.1 `void Matriz_Base::asignaNombre (const char * nmb)`

Asigna nombre a la matriz.

Parameters

| | |
|------------|---------------------|
| <i>nmb</i> | Nombre de la matriz |
|------------|---------------------|

6.4.3.2 `int Matriz_Base::columnas (void)` `[inline]`

Retorna el nmero de columnas de la matriz.

Returns

Regresa el nmero de columnas de la matriz

6.4.3.3 `virtual int Matriz_Base::entradasDistintasCero (void)` `[pure virtual]`

Retorna el nmero de entradas distintas de cero en la matriz.

Returns

Nmero de entradas distintas de cero en la matriz

Implemented in [Matriz](#).

6.4.3.4 `void Matriz_Base::faltaMemoria (void)` `[inline]`

Visualiza el error de falta de memoria para soportar la matriz.

6.4.3.5 `virtual void Matriz_Base::liberaMemoria (void)` `[pure virtual]`

Libera la memoria solicitada para la matriz.

Implemented in [MatrizDensa](#), [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.4.3.6 `int Matriz_Base::matrizCuadrada(void)` `[inline]`

Es una matriz cuadrada.

Returns

(1) en caso de ser matriz cuadrada. (0) en caso contrario

6.4.3.7 `virtual int Matriz_Base::maximoTamanoColumnaOcupado(void)` `[pure virtual]`

Retorna el mmo tama todas las columnas que sean distintos de cero.

Returns

Mmo tama columnas ocupadas en la matriz

Implemented in [Matriz](#).

6.4.3.8 `int Matriz_Base::mismaDimension(Matriz_Base * a)` `[inline]`

Revisa si ambas matrices tienen la misma dimensi

Returns

(1) en caso de que ambas matrices tengan la misma dimensi0) en caso contrario

6.4.3.9 `char* Matriz_Base::nombre(void)` `[inline]`

Retorna el nombre de la matriz.

Returns

Regresa un puntero al nombre de la matriz

6.4.3.10 `int Matriz_Base::renglones(void)` `[inline]`

Retorna el nmero de renglones de la matriz.

Returns

Regresa el nmero de renglones de la matriz

6.4.3.11 `int Matriz_Base::retornaTipoMatriz(void)` `[inline]`

Retorna el tipo de matriz.

Returns

Tipo de matriz MATRIZ_BANDADA, MATRIZ_DENSA, MATRIZ_ENTERA_DENSA, MATRIZ_DISPERSA

6.4.3.12 `Idouble Matriz_Base::tamano(void)`

Tama la matriz (aproximado) en Kb.

Returns

Tama la matriz en Kb

6.4.3.13 `void Matriz_Base::visualizaInformacion(void)`

Visualiza informacineral de la matriz.

6.4.3.14 `void Matriz_Base::visualizaTamano(void)`

Visualiza el tama memoria de la matriz en Kb.

6.4.4 Member Data Documentation

6.4.4.1 `int Matriz_Base::Ban` `[protected]`

Tama la banda (solo si es bandada o dispersa)

6.4.4.2 `int Matriz_Base::Col` [protected]

Nmero de columnas.

6.4.4.3 `char* Matriz_Base::Nmb` [protected]

Nombre de la matriz.

6.4.4.4 `int Matriz_Base::Ren` [protected]

Nmero de renglones.

6.4.4.5 `int Matriz_Base::Tipo_Matriz` [protected]

Indica el tipo de matriz de que se trate.

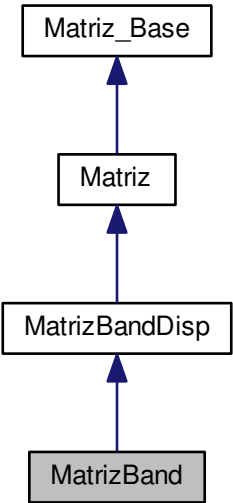
The documentation for this class was generated from the following files:

- [Matriz_Base.hpp](#)
- [Matriz_Base.cpp](#)

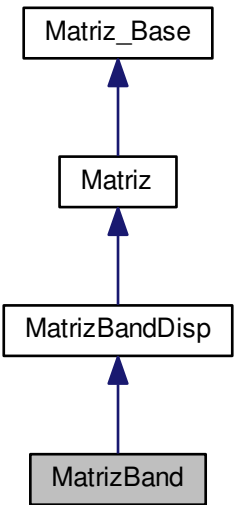
6.5 MatrizBand Class Reference

`#include <MatrizBand.hpp>`

Inheritance diagram for MatrizBand:



Collaboration diagram for MatrizBand:



Public Member Functions

- [MatrizBand](#) (const int ren, const int col, const int ban)
Constructor de la clase.
- [MatrizBand](#) (const int ren, const int col, const int ban, const char *nmb)
Constructor de la clase.
- [~MatrizBand](#) ()
- void [liberaMemoria](#) (void)
Libera la memoria solicitada para la matriz.
- void [inicializa](#) (ldouble val)
Inicializa la matriz al valor indicado.
- void [asigna](#) (const int ren, const int col, const ldouble val)
Asigna el valor indicado en el renglo y columna solicitado.
- ldouble [retorna](#) (const int ren, const int col)
Retorna el valor del renglon y columna solicitado.
- int [retornaNumeroColumnasBanda](#) (int ren)
Retorna el nmero de columnas de la banda para el renglon indicado.
- int [retornaNumeroColumna](#) (int ren, int ind)
Retorna el nmero de columna cuando se para en el renglon e indice de la banda.
- double [retornaValorColumna](#) (int ren, int ind)
Retorna el valor de la columna cuando se para en el renglon e indice de la banda.
- void [multiplica](#) (ldouble esc)
Multiplica la matriz por el escalar pasado como parametro.
- void [multiplica](#) (MatrizBand *a, MatrizBand *b)
Multiplica la matriz A por la matriz B.
- void [multiplica](#) (Vector *b, Vector *r)
Multiplica la matriz por el vector B dejando el Resultado en R.
- void [visualizaMatricesInternas](#) (void)
Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Additional Inherited Members

6.5.1 Detailed Description

Clase para el trabajar con matrices bandadas de punto flotante basada en el algoritmo Compressed Diagonal Storage (CDS) El algoritmo esta optimizado para hacer producto matriz vector

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

Todo Hacer comportamiento para cambiar tama banda
Multiplicaci matrices

6.5.2 Constructor & Destructor Documentation

6.5.2.1 MatrizBand::MatrizBand (const int *ren*, const int *col*, const int *ban*) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>ban</i> | Tama la banda |

6.5.2.2 MatrizBand::MatrizBand (const int *ren*, const int *col*, const int *ban*, const char * *nmb*) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>ban</i> | Tama la banda |
| <i>nmb</i> | Nombre de la matriz |

6.5.2.3 MatrizBand::~~MatrizBand () [inline]

6.5.3 Member Function Documentation

6.5.3.1 void MatrizBand::asigna (const int *ren*, const int *col*, const Idouble *val*) [virtual]

Asigna el valor indicado en el renglo y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |
| <i>val</i> | Valor |

Implements [Matriz](#).

6.5.3.2 void MatrizBand::inicializa (Idouble *val*) [inline],[virtual]

Inicializa la matriz al valor indicado.

Parameters

| | |
|------------|---|
| <i>val</i> | Valor por omisira inicializar la matriz |
|------------|---|

Implements [Matriz](#).

6.5.3.3 void MatrizBand::liberaMemoria (void) [virtual]

Libera la memoria solicitada para la matriz.

Implements [Matriz_Base](#).

6.5.3.4 void MatrizBand::multiplica (Idouble *esc*) [virtual]

Multiplica la matriz por el escalar pasado como parametro.

Parameters

| | |
|------------|---------|
| <i>esc</i> | Escalar |
|------------|---------|

Implements [MatrizBandDisp](#).

6.5.3.5 void MatrizBand::multiplica (MatrizBand * a, MatrizBand * b)

Multiplica la matriz A por la matriz B.

Parameters

| | |
|----------|------------------------|
| <i>a</i> | Puntero a matriz densa |
| <i>b</i> | Puntero a matriz densa |

6.5.3.6 void MatrizBand::multiplica (Vector * b, Vector * r) [virtual]

Multiplica la matriz por el vector B dejando el Resultado en R.

Parameters

| | |
|----------|-------------------------------------|
| <i>b</i> | Puntero a un Vector |
| <i>r</i> | Puntero a un Vector |

Implements [MatrizBandDisp](#).

6.5.3.7 Idouble MatrizBand::retorna (const int ren, const int col) [virtual]

Retorna el valor del renglon y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |

Returns

Valor

Implements [Matriz](#).

6.5.3.8 int MatrizBand::retornaNumeroColumna (int ren, int ind) [virtual]

Retorna el nmero de columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Nmero de columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.5.3.9 int MatrizBand::retornaNumeroColumnasBanda (int ren) [virtual]

Retorna el nmero de columnas de la banda para el renglon indicado.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
|------------|------------------|

Returns

Nmero de columnas de la banda para el renglon solicitado

Implements [Matriz](#).

6.5.3.10 double MatrizBand::retornaValorColumna (int ren, int ind) [inline],[virtual]

Retorna el valor de la columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Valor de la columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.5.3.11 void MatrizBand::visualizaMatricesInternas (void) [virtual]

Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Implements [MatrizBandDisp](#).

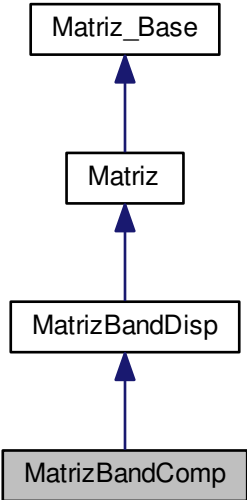
The documentation for this class was generated from the following files:

- [MatrizBand.hpp](#)
- [MatrizBand.cpp](#)

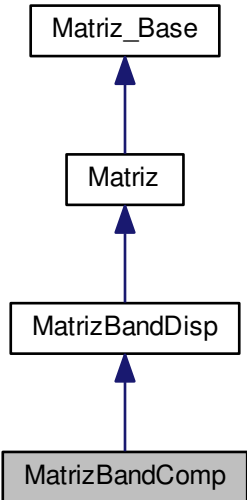
6.6 MatrizBandComp Class Reference

```
#include <MatrizBandComp.hpp>
```

Inheritance diagram for MatrizBandComp:



Collaboration diagram for MatrizBandComp:



Public Member Functions

- [MatrizBandComp](#) (const int ren, const int col, const int ban)
Constructor de la clase.
- [MatrizBandComp](#) (const int ren, const int col, const int ban, const char *nmb)
Constructor de la clase.
- [~MatrizBandComp](#) ()
Destructor de la clase.
- void [liberaMemoria](#) (void)
Libera la memoria solicitada para la matriz.
- void [inicializa](#) (ldouble val)
Inicializa la matriz al valor indicado.
- void [asigna](#) (const int ren, const int col, const ldouble val)
Asigna el valor indicado en el renglo y columna solicitado.
- ldouble [retorna](#) (const int ren, const int col)
Retorna el valor del renglon y columna solicitado.
- int [retornaNumeroColumnasBanda](#) (int ren)
Retorna el nmero de columnas de la banda para el renglon indicado.
- int [retornaNumeroColumna](#) (int ren, int ind)
Retorna el nmero de columna cuando se para en el renglon e indice de la banda.
- double [retornaValorColumna](#) (int ren, int ind)
Retorna el valor de la columna cuando se para en el renglon e indice de la banda.
- void [multiplica](#) (ldouble esc)
Multiplica la matriz por el escalar pasado como parametro.
- void [multiplica](#) (MatrizBandComp *a, MatrizBandComp *b)
Multiplica la matriz A por la matriz B.
- void [multiplica](#) (Vector *b, Vector *r)
Multiplica la matriz por el vector B dejando el Resultado en R.
- void [visualizaMatricesInternas](#) (void)
Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Additional Inherited Members

6.6.1 Detailed Description

Clase para el trabajar con matrices bandadas de punto flotante basada en el algoritmo Compressed Diagonal Storage (CDS) que pueden ser compactada, pero soporta matrices con todos los valores distintos. El algoritmo esta optimizado para hacer el producto matriz vector. El algoritmo de compactaciesupone que se llenara la matriz renglon por renglon de manara consecutiva, no asi las columnas dentro del renglon. El algoritmo de compactacilo guarda los renglones distintos de la matriz. Si la matriz tiene ciertas propiedades es posible usar otros algoritmo de compactaciatrices tridiagonales con coeficientes constantes solo guardar los 3 valores distintos).

Author

Antonio Carrillo Ledesma

Date

verano 2009

Version

1.0.1

- Bug** No hay errores conocidos
- Todo** Multiplicaci matrices

6.6.2 Constructor & Destructor Documentation

6.6.2.1 MatrizBandComp::MatrizBandComp (const int ren, const int col, const int ban) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>ban</i> | Tama la banda |

6.6.2.2 MatrizBandComp::MatrizBandComp (const int ren, const int col, const int ban, const char * nmb) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>ban</i> | Tama la banda |
| <i>nmb</i> | Nombre de la matriz |

6.6.2.3 MatrizBandComp::~MatrizBandComp () [inline]

Destructor de la clase.

6.6.3 Member Function Documentation

6.6.3.1 void MatrizBandComp::asigna (const int *ren*, const int *col*, const Idouble *val*) [virtual]

Asigna el valor indicado en el renglo y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |
| <i>val</i> | Valor |

Implements [Matriz](#).

6.6.3.2 void MatrizBandComp::inicializa (Idouble *val*) [inline],[virtual]

Inicializa la matriz al valor indicado.

Parameters

| | |
|------------|---|
| <i>val</i> | Valor por omisira inicializar la matriz |
|------------|---|

Implements [Matriz](#).

6.6.3.3 void MatrizBandComp::liberaMemoria (void) [virtual]

Libera la memoria solicitada para la matriz.

Implements [Matriz_Base](#).

6.6.3.4 void MatrizBandComp::multiplica (Idouble *esc*) [virtual]

Multiplica la matriz por el escalar pasado como parametro.

Parameters

| | |
|------------|---------|
| <i>esc</i> | Escalar |
|------------|---------|

Implements [MatrizBandDisp](#).

6.6.3.5 void MatrizBandComp::multiplica (MatrizBandComp * *a*, MatrizBandComp * *b*)

Multiplica la matriz A por la matriz B.

Parameters

| | |
|----------|------------------------|
| <i>a</i> | Puntero a matriz densa |
| <i>b</i> | Puntero a matriz densa |

6.6.3.6 void MatrizBandComp::multiplica (Vector * *b*, Vector * *r*) [virtual]

Multiplica la matriz por el vector B dejando el Resultado en R.

Parameters

| | |
|----------|-------------------------------------|
| <i>b</i> | Puntero a un Vector |
| <i>r</i> | Puntero a un Vector |

Implements [MatrizBandDisp](#).

6.6.3.7 Idouble MatrizBandComp::retorna (const int *ren*, const int *col*) [virtual]

Retorna el valor del renglon y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |

Returns

Valor

Implements [Matriz](#).

6.6.3.8 `int MatrizBandComp::retornaNumeroColumna (int ren, int ind)` `[virtual]`

Retorna el nmero de columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Nmero de columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.6.3.9 `int MatrizBandComp::retornaNumeroColumnasBanda (int ren)` `[virtual]`

Retorna el nmero de columnas de la banda para el renglon indicado.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
|------------|------------------|

Returns

Nmero de columnas de la banda para el renglon solicitado

Implements [Matriz](#).

6.6.3.10 `double MatrizBandComp::retornaValorColumna (int ren, int ind)` `[inline],[virtual]`

Retorna el valor de la columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Valor de la columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.6.3.11 `void MatrizBandComp::visualizaMatricesInternas (void)` `[virtual]`

Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Implements [MatrizBandDisp](#).

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

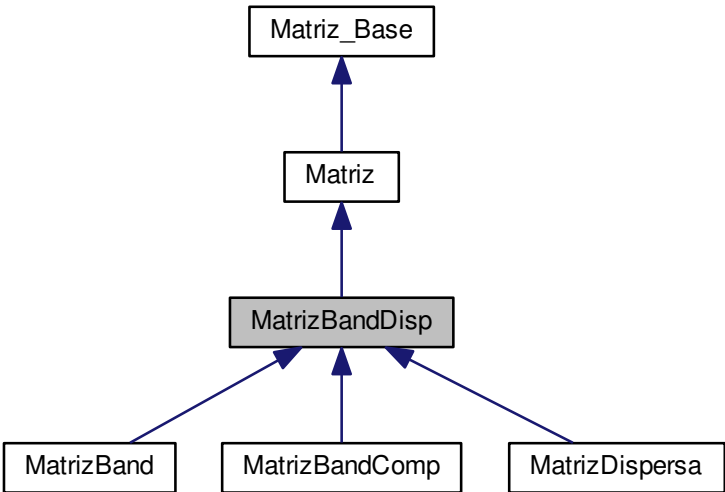
- [MatrizBandComp.hpp](#)
- [MatrizBandComp.cpp](#)

6.7 MatrizBandDisp Class Reference

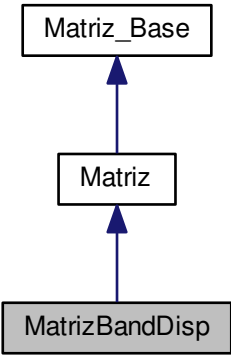
Clase base para trabajar con matrices bandadas y dispersas.

```
#include <MatrizBandDisp.hpp>
```

Inheritance diagram for MatrizBandDisp:



Collaboration diagram for MatrizBandDisp:



Public Member Functions

- MatrizBandDisp (void)
Constructor de la clase.
- int tamanoBanda (void)
Retorna el tama la banda.
- virtual void multiplica (ldouble esc)=0
Multiplica la matriz por el escalar pasado como parametro.
- virtual void multiplica (Vector *b, Vector *r)=0
Multiplica la matriz por el vector B dejando el Resultado en R.
- virtual void visualizaMatricesInternas (void)=0
Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Additional Inherited Members

6.7.1 Detailed Description

Clase base para trabajar con matrices bandadas y dispersas.

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.7.2 Constructor & Destructor Documentation

6.7.2.1 MatrizBandDisp::MatrizBandDisp (void) [inline]

Constructor de la clase.

6.7.3 Member Function Documentation

6.7.3.1 virtual void MatrizBandDisp::multiplica (Idouble esc) [pure virtual]

Multiplica la matriz por el escalar pasado como parametro.

Parameters

| | |
|-----|---------|
| esc | Escalar |
|-----|---------|

Implements [Matriz](#).

Implemented in [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.7.3.2 virtual void MatrizBandDisp::multiplica (Vector * b, Vector * r) [pure virtual]

Multiplica la matriz por el vector B dejando el Resultado en R.

Parameters

| | |
|---|-------------------------------------|
| b | Puntero a un Vector |
| r | Puntero a un Vector |

Implements [Matriz](#).

Implemented in [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

6.7.3.3 int MatrizBandDisp::tamanoBanda (void) [inline]

Retorna el tama la banda.

Returns

Tama la banda

6.7.3.4 virtual void MatrizBandDisp::visualizaMatricesInternas (void) [pure virtual]

Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Implemented in [MatrizBandComp](#), [MatrizDispersa](#), and [MatrizBand](#).

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

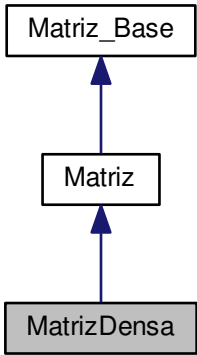
- [MatrizBandDisp.hpp](#)

6.8 MatrizDensa Class Reference

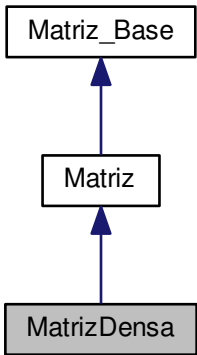
Clase para el trabajar con matrices densas de punto flotante.

#include <MatrizDensa.hpp>

Inheritance diagram for MatrizDensa:



Collaboration diagram for MatrizDensa:



Public Member Functions

- MatrizDensa (void)
Constructor de la clase.
- MatrizDensa (MatrizDensa &mat)
Constructor de la clase.
- MatrizDensa (const int ren, const int col)
Constructor de la clase.
- MatrizDensa (const int ren, const int col, const char *nmb)
Constructor de la clase.
- ~MatrizDensa ()
Destructor de la clase.
- void liberaMemoria (void)
Libera la memoria solicitada para la matriz.
- void redimensiona (int nren, int ncol)
Constructor de la clase.
- void inicializa (const Idouble val)
Inicializa la matriz al valor indicado.
- void asigna (const int ren, const int col, const Idouble val)
Asigna el valor indicado en el renglo y columna solicitado.
- Idouble retorna (const int ren, const int col)
Retorna el valor del renglon y columna solicitado.
- void multiplica (Idouble esc)
Multiplica la matriz por el escalar pasado como parametro.
- void multiplica (MatrizDensa *a, MatrizDensa *b)
Multiplica la matriz A por la matriz B.
- void multiplica (Vector *b, Vector *r)
Multiplica la matriz por el vector B dejando el Resultado en R.
- int retornaNumeroColumnasBanda (int ren)
Retorna el nmero de columnas de la banda para el renglon indicado.

- int [retornaNumeroColumna](#) (int ren, int ind)
Retorna el nmero de columna cuando se para en el renglon e indice de la banda.
- double [retornaValorColumna](#) (int ren, int ind)
Retorna el valor de la columna cuando se para en el renglon e indice de la banda.

Additional Inherited Members

6.8.1 Detailed Description

Clase para el trabajar con matrices densas de punto flotante.

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.8.2 Constructor & Destructor Documentation

6.8.2.1 MatrizDensa::MatrizDensa (void) [inline]

Constructor de la clase.

6.8.2.2 MatrizDensa::MatrizDensa (MatrizDensa & mat) [inline]

Constructor de la clase.

Genera una matriz a partir de los datos de otra

Parameters

| | |
|------------|---|
| <i>mat</i> | Matriz del cual se hara una copia |
|------------|---|

6.8.2.3 MatrizDensa::MatrizDensa (const int ren, const int col) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |

6.8.2.4 MatrizDensa::MatrizDensa (const int ren, const int col, const char * nmb) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>nmb</i> | Nombre de la matriz |

6.8.2.5 MatrizDensa::~~MatrizDensa () [inline]

Destructor de la clase.

6.8.3 Member Function Documentation

6.8.3.1 void MatrizDensa::asigna (const int ren, const int col, const ldouble val) [virtual]

Asigna el valor indicado en el renglo y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |
| <i>val</i> | Valor |

Implements [Matriz](#).

6.8.3.2 void MatrizDensa::inicializa (const Idouble *val*) [virtual]

Inicializa la matriz al valor indicado.

Parameters

| | |
|------------|---|
| <i>val</i> | Valor por omisira inicializar la matriz |
|------------|---|

Implements [Matriz](#).

6.8.3.3 void MatrizDensa::liberaMemoria (void) [virtual]

Libera la memoria solicitada para la matriz.

Implements [Matriz_Base](#).

6.8.3.4 void MatrizDensa::multiplica (Idouble *esc*) [virtual]

Multiplica la matriz por el escalar pasado como parametro.

Parameters

| | |
|------------|---------|
| <i>esc</i> | Escalar |
|------------|---------|

Implements [Matriz](#).

6.8.3.5 void MatrizDensa::multiplica (MatrizDensa * *a*, MatrizDensa * *b*)

Multiplica la matriz A por la matriz B.

Parameters

| | |
|----------|------------------------|
| <i>a</i> | Puntero a matriz densa |
| <i>b</i> | Puntero a matriz densa |

6.8.3.6 void MatrizDensa::multiplica (Vector * *b*, Vector * *r*) [virtual]

Multiplica la matriz por el vector B dejando el Resultado en R.

Parameters

| | |
|----------|-------------------------------------|
| <i>b</i> | Puntero a un Vector |
| <i>r</i> | Puntero a un Vector |

Implements [Matriz](#).

6.8.3.7 void MatrizDensa::redimensiona (int *nren*, int *ncol*) [inline]

Constructor de la clase.

Parameters

| | |
|-------------|---------------------------------------|
| <i>nren</i> | Nuevo nmero de renglones de la matriz |
| <i>ncol</i> | Nuevo nmero de columnas de la matriz |

6.8.3.8 Idouble MatrizDensa::retorna (const int *ren*, const int *col*) [virtual]

Retorna el valor del renglon y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |

Returns

Valor

Implements [Matriz](#).

6.8.3.9 `int MatrizDensa::retornaNumeroColumna (int ren, int ind)` `[inline],[virtual]`

Retorna el nmero de columna cuando se para en el renglon e índice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Nmero de columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.8.3.10 `int MatrizDensa::retornaNumeroColumnasBanda (int ren)` `[inline],[virtual]`

Retorna el nmero de columnas de la banda para el renglon indicado.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
|------------|------------------|

Returns

Nmero de columnas de la banda para el renglon solicitado

Implements [Matriz](#).

6.8.3.11 `double MatrizDensa::retornaValorColumna (int ren, int ind)` `[inline],[virtual]`

Retorna el valor de la columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Valor de la columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

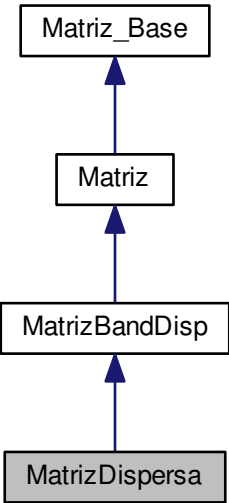
The documentation for this class was generated from the following files:

- [MatrizDensa.hpp](#)
- [MatrizDensa.cpp](#)

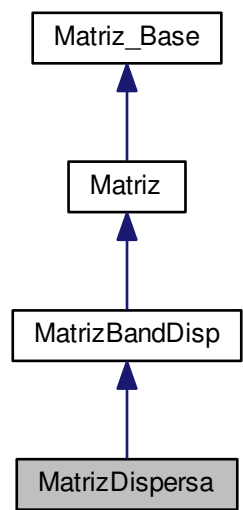
6.9 MatrizDispersa Class Reference

```
#include <MatrizDispersa.hpp>
```

Inheritance diagram for MatrizDispersa:



Collaboration diagram for MatrizDispersa:



Public Member Functions

- [MatrizDispersa](#) (const int ren, const int col, const int ban)
Constructor de la clase.
- [MatrizDispersa](#) (const int ren, const int col, const int ban, const char *nmb)
Constructor de la clase.
- [~MatrizDispersa](#) ()
- void [liberaMemoria](#) (void)
Libera la memoria solicitada para la matriz.
- void [inicializa](#) (ldouble val)
Inicializa la matriz al valor indicado.
- void [asigna](#) (const int ren, const int col, const ldouble val)
Asigna el valor indicado en el renglo y columna solicitado.
- ldouble [retorna](#) (const int ren, const int col)
Retorna el nmero de columna cuando se para en el renglon e indice de la banda.
- int [retornaNumeroColumnasBanda](#) (int ren)
Retorna el nmero de columnas de la banda para el renglon indicado.
- int [retornaNumeroColumna](#) (int ren, int ind)
Retorna el nmero de columna cuando se para en el renglon e indice de la banda.
- double [retornaValorColumna](#) (int ren, int ind)
Retorna el valor de la columna cuando se para en el renglon e indice de la banda.
- void [multiplica](#) (ldouble esc)
Multiplica la matriz por el escalar pasado como parametro.
- void [multiplica](#) (MatrizDispersa *a, MatrizDispersa *b)
Multiplica la matriz A por la matriz B.
- void [multiplica](#) (Vector *b, Vector *r)
Multiplica la matriz por el vector B dejando el Resultado en R.
- void [visualizaMatricesInternas](#) (void)
Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Additional Inherited Members

6.9.1 Detailed Description

Clase para el trabajar con matrices dispersas de punto flotante basada en el algoritmo Jagged Diagonal Storage (JDS) El algoritmo esta optimizado para hacer producto matriz vector

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

Todo Hacer comportamiento para cambiar tama banda
Multiplicaci matrices

6.9.2 Constructor & Destructor Documentation

6.9.2.1 MatrizDispersa::MatrizDispersa (const int *ren*, const int *col*, const int *ban*) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>ban</i> | Tama la banda |

6.9.2.2 MatrizDispersa::MatrizDispersa (const int *ren*, const int *col*, const int *ban*, const char * *nmb*) [inline]

Constructor de la clase.

Parameters

| | |
|------------|---------------------------------|
| <i>ren</i> | Nmero de renglones de la matriz |
| <i>col</i> | Nmero de columnas de la matriz |
| <i>ban</i> | Tama la banda |
| <i>nmb</i> | Nombre de la matriz |

6.9.2.3 MatrizDispersa::~~MatrizDispersa () [inline]

6.9.3 Member Function Documentation

6.9.3.1 void MatrizDispersa::asigna (const int *ren*, const int *col*, const Idouble *val*) [virtual]

Asigna el valor indicado en el renglo y columna solicitado.

Parameters

| | |
|------------|---------|
| <i>ren</i> | Renglon |
| <i>col</i> | Columna |
| <i>val</i> | Valor |

Implements [Matriz](#).

6.9.3.2 void MatrizDispersa::inicializa (Idouble *val*) [inline],[virtual]

Inicializa la matriz al valor indicado.

Parameters

| | |
|------------|---|
| <i>val</i> | Valor por omisira inicializar la matriz |
|------------|---|

Implements [Matriz](#).

6.9.3.3 void MatrizDispersa::liberaMemoria (void) [virtual]

Libera la memoria solicitada para la matriz.

Implements [Matriz_Base](#).

6.9.3.4 void MatrizDispersa::multiplica (Idouble *esc*) [virtual]

Multiplica la matriz por el escalar pasado como parametro.

Parameters

| | |
|------------|---------|
| <i>esc</i> | Escalar |
|------------|---------|

Implements [MatrizBandDisp](#).

6.9.3.5 void MatrizDispersa::multiplica (MatrizDispersa * a, MatrizDispersa * b)

Multiplica la matriz A por la matriz B.

Parameters

| | |
|---|------------------------|
| a | Puntero a matriz densa |
| b | Puntero a matriz densa |

6.9.3.6 void MatrizDispersa::multiplica (Vector * b, Vector * r) [virtual]

Multiplica la matriz por el vector B dejando el Resultado en R.

Parameters

| | |
|---|-------------------------------------|
| b | Puntero a un Vector |
| r | Puntero a un Vector |

Implements [MatrizBandDisp](#).

6.9.3.7 Idouble MatrizDispersa::retorna (const int ren, const int col) [virtual]

Retorna el nmero de columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|-----|------------------|
| ren | Nmero de renglon |
| col | Nmero de columna |

Returns

Nmero de columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.9.3.8 int MatrizDispersa::retornaNumeroColumna (int ren, int ind) [inline],[virtual]

Retorna el nmero de columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|-----|------------------|
| ren | Nmero de renglon |
| ind | Nmero de indice |

Returns

Nmero de columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.9.3.9 int MatrizDispersa::retornaNumeroColumnasBanda (int ren) [virtual]

Retorna el nmero de columnas de la banda para el renglon indicado.

Parameters

| | |
|-----|------------------|
| ren | Nmero de renglon |
|-----|------------------|

Returns

Nmero de columnas de la banda para el renglon solicitado

Implements [Matriz](#).

6.9.3.10 double MatrizDispersa::retornaValorColumna (int ren, int ind) [inline],[virtual]

Retorna el valor de la columna cuando se para en el renglon e indice de la banda.

Parameters

| | |
|------------|------------------|
| <i>ren</i> | Nmero de renglon |
| <i>ind</i> | Nmero de indice |

Returns

Valor de la columna cuando se para en el renglon e indice de la banda

Implements [Matriz](#).

6.9.3.11 void MatrizDispersa::visualizaMatricesInternas (void) [virtual]

Visualiza las matrices internas usadas para soportar a las matrices bandadas.

Implements [MatrizBandDisp](#).

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

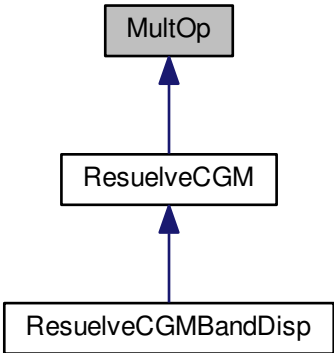
- [MatrizDispersa.hpp](#)
- [MatrizDispersa.cpp](#)

6.10 MultOp Class Reference

Clase para implementar la multiplicaci la matriz por el vector.

```
#include <MultOp.hpp>
```

Inheritance diagram for MultOp:



Public Member Functions

- virtual void [multiplica](#) ([Vector](#) *x, [Vector](#) *y)=0
 $y = A*x$
- virtual int [tamano](#) (void)=0
Tama

6.10.1 Detailed Description

Clase para implementar la multiplicaci la matriz por el vector.

Author

Antonio Carrillo Ledesma

Date

primavera 2010

Version

1.0.0

Bug No hay errores conocidos

6.10.2 Member Function Documentation

6.10.2.1 virtual void MultOp::multiplica ([Vector](#) * x, [Vector](#) * y) [pure virtual]

$y = A*x$

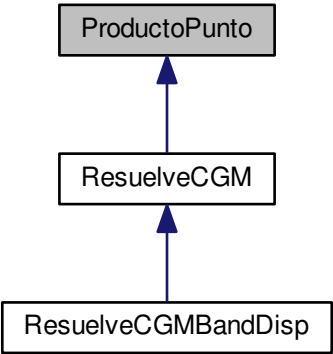
Implemented in [ResuelveCGM](#).

6.10.2.2 virtual int MultOp::tamano (void) [pure virtual]

Tama
Implemented in [ResuelveCGM](#).
The documentation for this class was generated from the following file:
* [MultOp.hpp](#)

6.11 ProductoPunto Class Reference

Clase para implementar el producto punto.
`#include <ProductoPunto.hpp>`
Inheritance diagram for ProductoPunto:



Public Member Functions

* virtual double [productoPunto](#) (Vector *x, Vector *y)=0
 Retorna el producto punto del vector X y Y.

6.11.1 Detailed Description

Clase para implementar el producto punto.
Author

Antonio Carrillo Ledesma

Date

primavera 2010

Version

1.0.0

Bug No hay errores conocidos

6.11.2 Member Function Documentation

6.11.2.1 virtual double ProductoPunto::productoPunto (Vector *x, Vector *y) [pure virtual]

Retorna el producto punto del vector X y Y.

Parameters

| | |
|---|---------------------|
| x | Puntero a un vector |
| y | Puntero a un vector |

Returns

El valor del producto punto

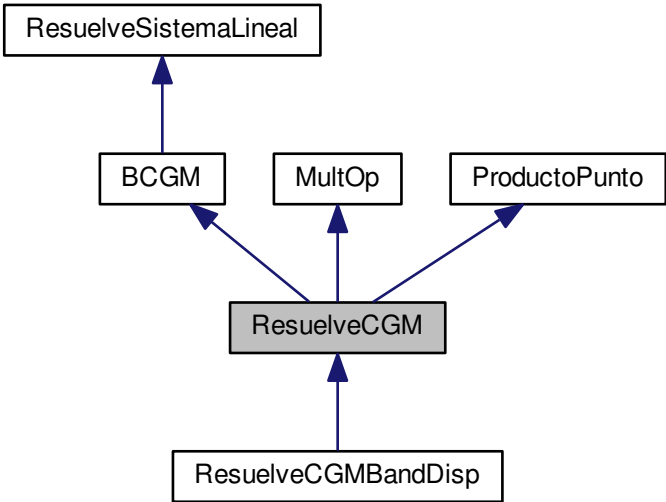
Implemented in [ResuelveCGM](#).
The documentation for this class was generated from the following file:
* [ProductoPunto.hpp](#)

6.12 ResuelveCGM Class Reference

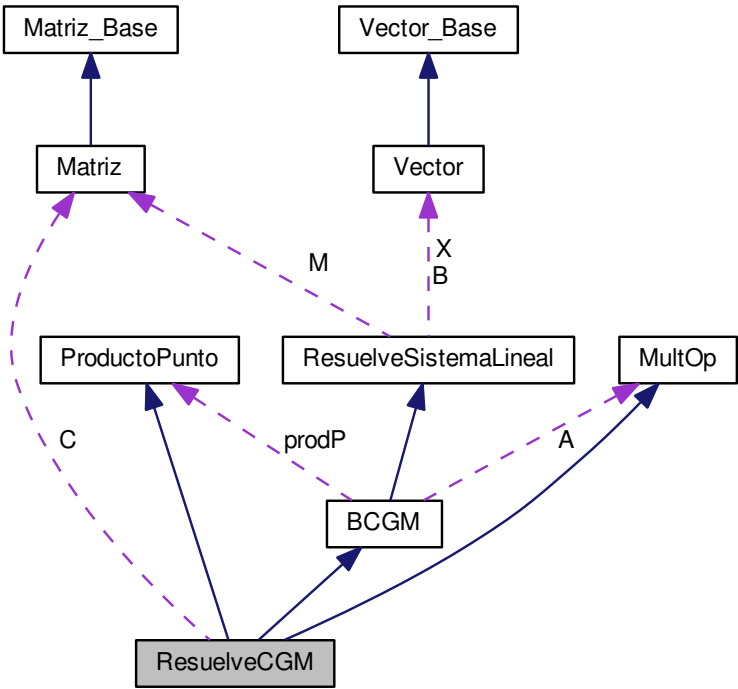
Clase para resolucil sistema lineal mediante CGM standard.

```
#include <ResuelveCGM.hpp>
```

Inheritance diagram for ResuelveCGM:



Collaboration diagram for ResuelveCGM:



Public Member Functions

- * [ResuelveCGM](#) (void)
Constructor de la clase.
- * [ResuelveCGM](#) (Vector *x, Vector *b)
Constructor de la clase.
- * [ResuelveCGM](#) (Vector *x, Vector *b, ldouble ep, int it)
Constructor de la clase.
- * void [configuraMetodo](#) (ldouble ep, int it)
Configura al mdo numco.

Protected Member Functions

- * double [productoPunto](#) ([Vector](#) *u, [Vector](#) *v)
Producto punto.
- * void [multiplica](#) ([Vector](#) *u, [Vector](#) *v)
Multiplica Au=v.
- * int [tamano](#) (void)
Tama

Protected Attributes

- [Matriz](#) * [C](#)
Precondicionador.

6.12.1 Detailed Description

Clase para resolucil sistema lineal mediante CGM standard.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.12.2 Constructor & Destructor Documentation

6.12.2.1 `ResuelveCGM::ResuelveCGM (void)` `[inline]`

Constructor de la clase.

6.12.2.2 `ResuelveCGM::ResuelveCGM (Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.12.2.3 `ResuelveCGM::ResuelveCGM (Vector * x, Vector * b, Idouble ep, int it)` `[inline]`

Constructor de la clase.
Parameters

| | |
|-----------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>ep</i> | Tolerancia del mdo |
| <i>it</i> | Nmero mmo de iteraciones |

6.12.3 Member Function Documentation

6.12.3.1 `void ResuelveCGM::configuraMetodo (Idouble ep, int it)` `[inline]`

Configura al mdo numco.
Parameters

| | |
|-----------|--------------------------|
| <i>ep</i> | Tolerancia del mdo |
| <i>it</i> | Nmero mmo de iteraciones |

6.12.3.2 `void ResuelveCGM::multiplica (Vector * u, Vector * v)` `[inline]`, `[protected]`, `[virtual]`

Multiplica Au=v.
Implements [MultOp](#).

6.12.3.3 `double ResuelveCGM::productoPunto (Vector * u, Vector * v)` `[inline]`, `[protected]`, `[virtual]`

Producto punto.
Implements [ProductoPunto](#).

6.12.3.4 `int ResuelveCGM::tamano (void) [inline],[protected],[virtual]`

Tama
Implements [MultOp](#).

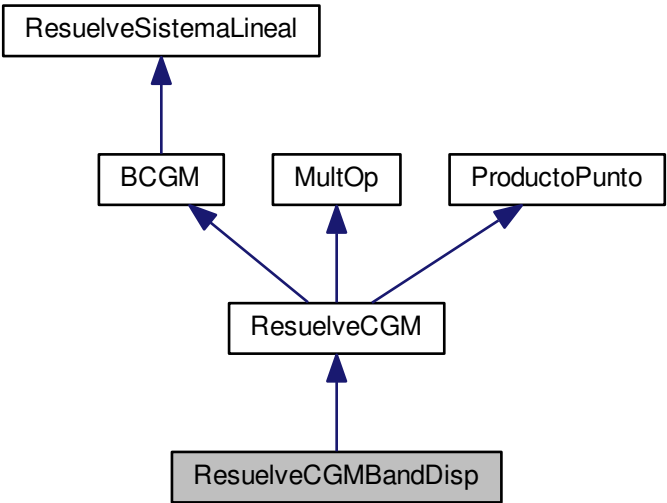
6.12.4 Member Data Documentation

6.12.4.1 `Matriz* ResuelveCGM::C [protected]`

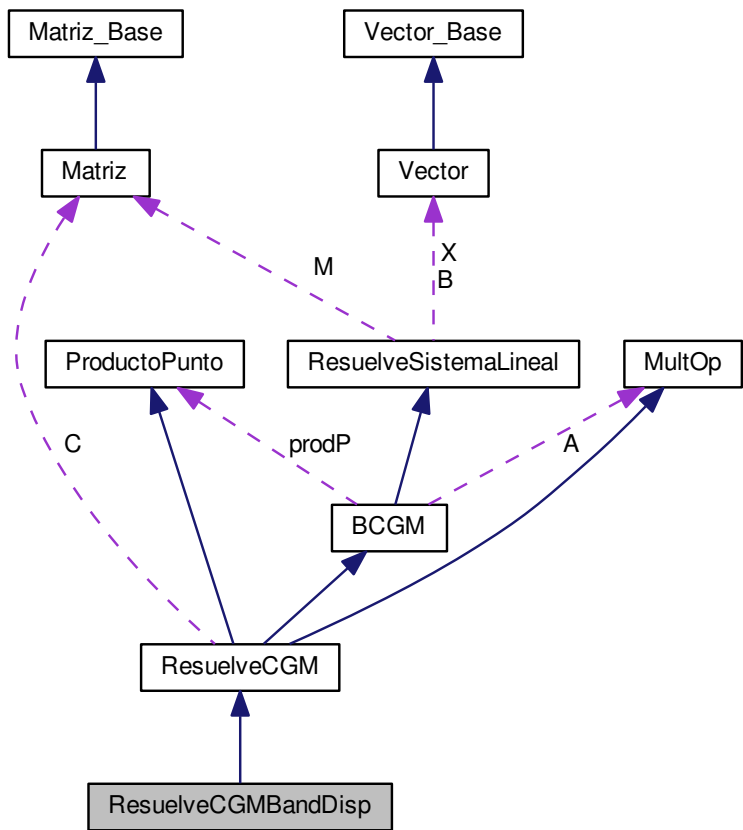
Precondicionador.
The documentation for this class was generated from the following file:
· [ResuelveCGM.hpp](#)

6.13 ResuelveCGMBandDisp Class Reference

Clase para resolucil sistema lineal mediante CGM.
`#include <ResuelveCGMBandDisp.hpp>`
Inheritance diagram for ResuelveCGMBandDisp:



Collaboration diagram for ResuelveCGMBandDisp:



Public Member Functions

- [ResuelveCGMBandDisp](#) (void)
Constructor de la clase.
- [ResuelveCGMBandDisp](#) (MatrizBandDisp *A)
Constructor de la clase.
- [ResuelveCGMBandDisp](#) (MatrizBandDisp *A, Vector *x, Vector *b)
Constructor de la clase.
- [ResuelveCGMBandDisp](#) (MatrizBandDisp *A, Vector *x, Vector *b, Idouble ep, int it)
Constructor de la clase.

Additional Inherited Members

6.13.1 Detailed Description

Clase para resolucil sistema lineal mediante CGM.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos
Todo Definir clase de producto interior y poder pasarlo como argumento y user este

6.13.2 Constructor & Destructor Documentation

6.13.2.1 `ResuelveCGMBandDisp::ResuelveCGMBandDisp (void)` `[inline]`
Constructor de la clase.

6.13.2.2 `ResuelveCGMBandDisp::ResuelveCGMBandDisp (MatrizBandDisp * A)` `[inline]`
Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
|----------|--|

6.13.2.3 `ResuelveCGMBandDisp::ResuelveCGMBandDisp (MatrizBandDisp * A, Vector * x, Vector * b)`
[inline]

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.13.2.4 `ResuelveCGMBandDisp::ResuelveCGMBandDisp (MatrizBandDisp * A, Vector * x, Vector * b, Idouble ep, int it)` [inline]

Constructor de la clase.
Parameters

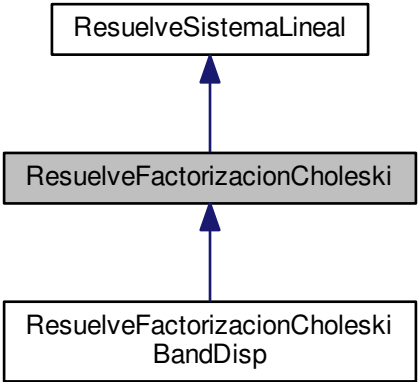
| | |
|-----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>ep</i> | Tolerancia del mdo |
| <i>it</i> | Nmero mmo de iteraciones |

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

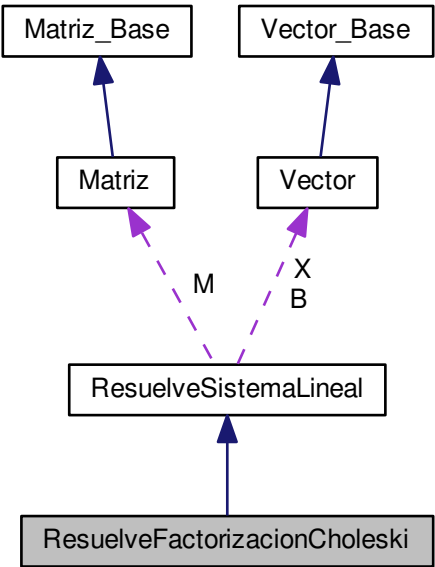
- [ResuelveCGMBandDisp.hpp](#)

6.14 ResuelveFactorizacionCholeski Class Reference

Clase para resolucil sistema lineal usando factorizacion Choleski.
`#include <ResuelveFactorizacionCholeski.hpp>`
Inheritance diagram for ResuelveFactorizacionCholeski:



Collaboration diagram for ResuelveFactorizacionCholeski:



Public Member Functions

- [ResuelveFactorizacionCholeski](#) (void)
Constructor de la clase.
- [ResuelveFactorizacionCholeski](#) ([Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- virtual void [factoriza](#) (void)=0
Factoriza la matriz A en L y U dejandolas en la misma matriz.

Protected Attributes

- bool [MatrizFactorizada](#)
[Matriz](#) factorizada.

Additional Inherited Members

6.14.1 Detailed Description

Clase para resolucil sistema lineal usando factorizacion Choleski.
Author

Antonio Carrillo Ledesma
Date

primavera 2010
Version

1.0.1
Bug No hay errores conocidos

6.14.2 Constructor & Destructor Documentation

6.14.2.1 `ResuelveFactorizacionCholeski::ResuelveFactorizacionCholeski (void) [inline]`

Constructor de la clase.

6.14.2.2 `ResuelveFactorizacionCholeski::ResuelveFactorizacionCholeski (Vector * x, Vector * b) [inline]`

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.14.3 Member Function Documentation

6.14.3.1 virtual void ResuelveFactorizacionCholeski::factoriza (void) [pure virtual]

Factoriza la matriz A en L y U dejandolas en la misma matriz.
Implemented in [ResuelveFactorizacionCholeskiBandDisp](#).

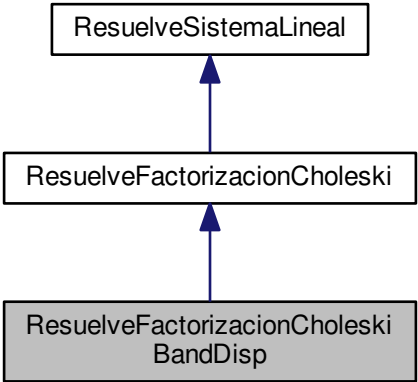
6.14.4 Member Data Documentation

6.14.4.1 bool ResuelveFactorizacionCholeski::MatrizFactorizada [protected]

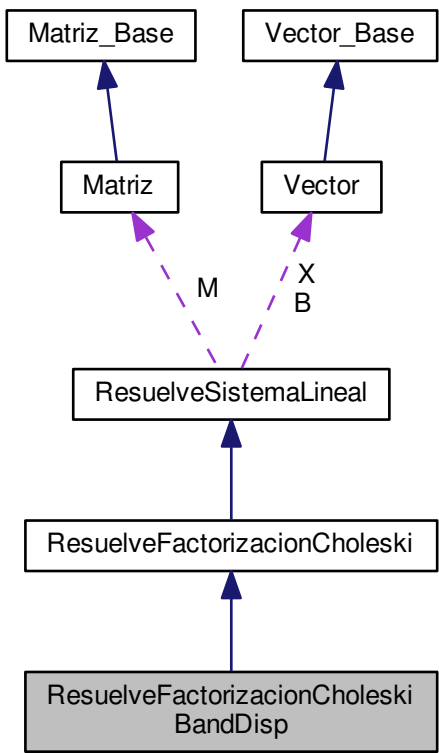
[Matriz](#) factorizada.
The documentation for this class was generated from the following file:
· [ResuelveFactorizacionCholeski.hpp](#)

6.15 ResuelveFactorizacionCholeskiBandDisp Class Reference

Clase para resolucil sistema lineal usando factorizacion Choleski.
`#include <ResuelveFactorizacionCholeskiBandDisp.hpp>`
Inheritance diagram for ResuelveFactorizacionCholeskiBandDisp:



Collaboration diagram for ResuelveFactorizacionCholeskiBandDisp:



Public Member Functions

- [ResuelveFactorizacionCholeskiBandDisp](#) (void)
Constructor de la clase.
- [ResuelveFactorizacionCholeskiBandDisp](#) (MatrizBandDisp *A)
Constructor de la clase.
- [ResuelveFactorizacionCholeskiBandDisp](#) (MatrizBandDisp *A, Vector *x, Vector *b)
Constructor de la clase.
- [ResuelveFactorizacionCholeskiBandDisp](#) (MatrizBand *A, Vector *x, Vector *b)
Constructor de la clase.
- [~ResuelveFactorizacionCholeskiBandDisp](#) ()
- void [factoriza](#) (void)
Factoriza la matriz A en L y U dejandolas en la misma matriz.
- void [resuelve](#) (void)
Resuelve el sistema lineal.
- void [resuelve](#) (Vector *x, Vector *b)
Resuelve el sistema lineal.

Additional Inherited Members

6.15.1 Detailed Description

Clase para resolucil sistema lineal usando factorizacion Choleski.

Author

Antonio Carrillo Ledesma

Date

primavera 2010

Version

1.0.1

Bug No hay errores conocidos

6.15.2 Constructor & Destructor Documentation

6.15.2.1 [ResuelveFactorizacionCholeskiBandDisp::ResuelveFactorizacionCholeskiBandDisp](#) (void)
[inline]

Constructor de la clase.

6.15.2.2 [ResuelveFactorizacionCholeskiBandDisp::ResuelveFactorizacionCholeskiBandDisp](#) (MatrizBandDisp * A) [inline]

Constructor de la clase.

Parameters

| | |
|-----|--|
| A | Puntero a una matriz del tipo MatrizBandDisp |
|-----|--|

6.15.2.3 `ResuelveFactorizacionCholeskiBandDisp::ResuelveFactorizacionCholeskiBandDisp (MatrizBandDisp * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|-----|--|
| A | Puntero a una matriz del tipo MatrizBandDisp |
| x | Puntero a un Vector , solucil sistema lineal |
| b | Puntero a un vector, lado derecho del sistema lineal |

6.15.2.4 `ResuelveFactorizacionCholeskiBandDisp::ResuelveFactorizacionCholeskiBandDisp (MatrizBand * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|-----|--|
| A | Puntero a una matriz del tipo MatrizBandDisp |
| x | Puntero a un Vector , solucil sistema lineal |
| b | Puntero a un vector, lado derecho del sistema lineal |

6.15.2.5 `ResuelveFactorizacionCholeskiBandDisp::~ResuelveFactorizacionCholeskiBandDisp ()` `[inline]`

6.15.3 Member Function Documentation

6.15.3.1 `void ResuelveFactorizacionCholeskiBandDisp::factoriza (void)` `[virtual]`

Factoriza la matriz A en L y U dejandolas en la misma matriz.
Implements [ResuelveFactorizacionCholeski](#).

6.15.3.2 `void ResuelveFactorizacionCholeskiBandDisp::resuelve (void)` `[virtual]`

Resuelve el sistema lineal.
Implements [ResuelveSistemaLineal](#).

6.15.3.3 `void ResuelveFactorizacionCholeskiBandDisp::resuelve (Vector * x, Vector * b)` `[inline]`, `[virtual]`

Resuelve el sistema lineal.
Parameters

| | |
|-----|--|
| x | Puntero a un vector, solucil sistema lineal |
| b | Puntero a un vector, lado derecho del sistema lineal |

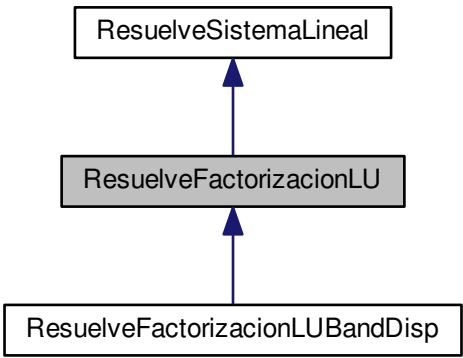
Implements [ResuelveSistemaLineal](#).
The documentation for this class was generated from the following files:

- [ResuelveFactorizacionCholeskiBandDisp.hpp](#)
- [ResuelveFactorizacionCholeskiBandDisp.cpp](#)

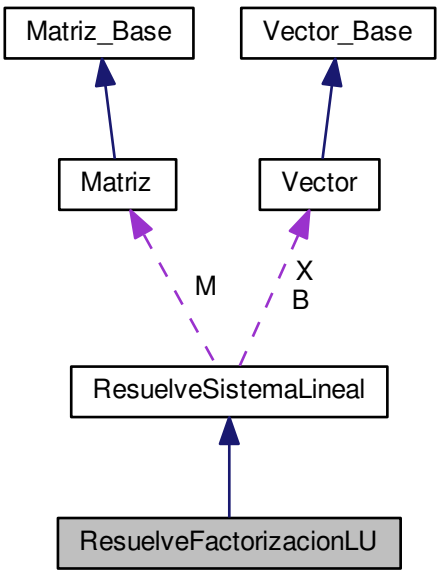
6.16 ResuelveFactorizacionLU Class Reference

Clase para resolucil sistema lineal usando factorizacion LU.
`#include <ResuelveFactorizacionLU.hpp>`

Inheritance diagram for ResuelveFactorizacionLU:



Collaboration diagram for ResuelveFactorizacionLU:



Public Member Functions

- [ResuelveFactorizacionLU](#) (void)
Constructor de la clase.
- [ResuelveFactorizacionLU](#) ([Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- virtual void [factoriza](#) (void)=0
Factoriza la matriz A en L y U dejandolas en la misma matriz.

Protected Attributes

- bool [MatrizFactorizada](#)
Matriz factorizada.

Additional Inherited Members

6.16.1 Detailed Description

Clase para resolucil sistema lineal usando factorizacion LU.
Author

Antonio Carrillo Ledesma
Date
primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.16.2 Constructor & Destructor Documentation

6.16.2.1 ResuelveFactorizacionLU::ResuelveFactorizacionLU (void) [inline]

Constructor de la clase.

6.16.2.2 ResuelveFactorizacionLU::ResuelveFactorizacionLU (Vector * x, Vector * b) [inline]

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.16.3 Member Function Documentation

6.16.3.1 virtual void ResuelveFactorizacionLU::factoriza (void) [pure virtual]

Factoriza la matriz A en L y U dejandolas en la misma matriz.
Implemented in [ResuelveFactorizacionLUBandDisp](#).

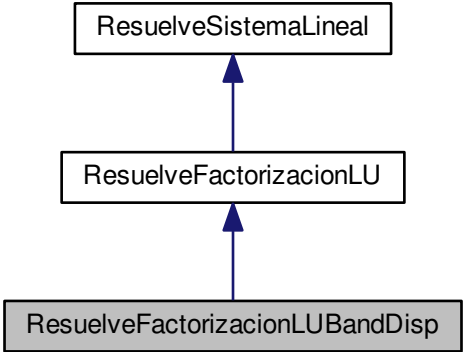
6.16.4 Member Data Documentation

6.16.4.1 bool ResuelveFactorizacionLU::MatrizFactorizada [protected]

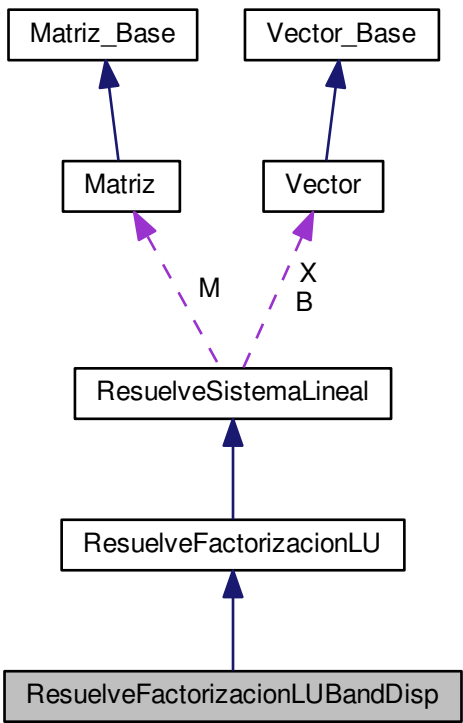
[Matriz](#) factorizada.
The documentation for this class was generated from the following file:
· [ResuelveFactorizacionLU.hpp](#)

6.17 ResuelveFactorizacionLUBandDisp Class Reference

Clase para resolucil sistema lineal usando factorizacion LU.
`#include <ResuelveFactorizacionLUBandDisp.hpp>`
Inheritance diagram for ResuelveFactorizacionLUBandDisp:



Collaboration diagram for ResuelveFactorizacionLUBandDisp:



Public Member Functions

- [ResuelveFactorizacionLUBandDisp](#) (void)
Constructor de la clase.
- [ResuelveFactorizacionLUBandDisp](#) ([MatrizBandDisp](#) *A)
Constructor de la clase.
- [ResuelveFactorizacionLUBandDisp](#) ([MatrizBandDisp](#) *A, [Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- [ResuelveFactorizacionLUBandDisp](#) ([MatrizBand](#) *A, [Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- [~ResuelveFactorizacionLUBandDisp](#) ()
- void [factoriza](#) (void)
Factoriza la matriz A en L y U dejandolas en la misma matriz.
- void [resuelve](#) (void)
Resuelve el sistema lineal.
- void [resuelve](#) ([Vector](#) *x, [Vector](#) *b)
Resuelve el sistema lineal.

Protected Attributes

- bool [liberaMatriz](#)
Indica si la matriz debera ser borrada.

Additional Inherited Members

6.17.1 Detailed Description

Clase para resolucil sistema lineal usando factorizacion LU.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.17.2 Constructor & Destructor Documentation

6.17.2.1 [ResuelveFactorizacionLUBandDisp::ResuelveFactorizacionLUBandDisp \(void \)](#) `[inline]`

Constructor de la clase.

6.17.2.2 ResuelveFactorizacionLUBandDisp::ResuelveFactorizacionLUBandDisp (MatrizBandDisp * A)
[inline]

Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
|----------|--|

6.17.2.3 `ResuelveFactorizacionLUBandDisp::ResuelveFactorizacionLUBandDisp (MatrizBandDisp * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.17.2.4 `ResuelveFactorizacionLUBandDisp::ResuelveFactorizacionLUBandDisp (MatrizBand * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.17.2.5 `ResuelveFactorizacionLUBandDisp::~~ResuelveFactorizacionLUBandDisp ()` `[inline]`

6.17.3 Member Function Documentation

6.17.3.1 `void ResuelveFactorizacionLUBandDisp::factoriza (void)` `[virtual]`

Factoriza la matriz A en L y U dejandolas en la misma matriz.
Implements [ResuelveFactorizacionLU](#).

6.17.3.2 `void ResuelveFactorizacionLUBandDisp::resuelve (void)` `[virtual]`

Resuelve el sistema lineal.
Implements [ResuelveSistemaLineal](#).

6.17.3.3 `void ResuelveFactorizacionLUBandDisp::resuelve (Vector * x, Vector * b)` `[inline]`,
`[virtual]`

Resuelve el sistema lineal.
Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un vector, solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

Implements [ResuelveSistemaLineal](#).

6.17.4 Member Data Documentation

6.17.4.1 `bool ResuelveFactorizacionLUBandDisp::liberaMatriz` `[protected]`

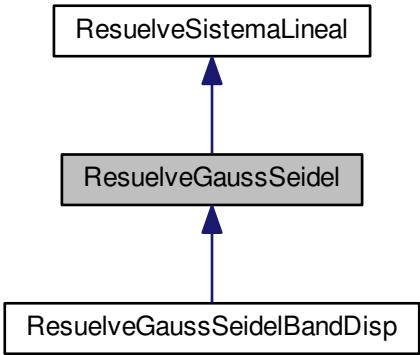
Indica si la matriz debera ser borrada.
The documentation for this class was generated from the following files:

- [ResuelveFactorizacionLUBandDisp.hpp](#)
- [ResuelveFactorizacionLUBandDisp.cpp](#)

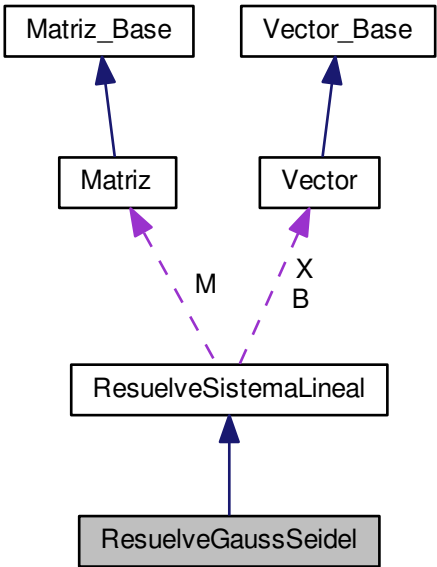
6.18 ResuelveGaussSeidel Class Reference

Clase para resolucil sistema lineal mediante Gauss-Seidel.
`#include <ResuelveGaussSeidel.hpp>`

Inheritance diagram for ResuelveGaussSeidel:



Collaboration diagram for ResuelveGaussSeidel:



Public Member Functions

- [ResuelveGaussSeidel](#) (void)
Constructor de la clase.
- [ResuelveGaussSeidel](#) ([Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- [ResuelveGaussSeidel](#) ([Vector](#) *x, [Vector](#) *b, [ldouble](#) ep, int iter)

Protected Attributes

- [ldouble](#) Ep
Tolerancia.
- int [Iter](#)
Iteraciones mmas.

Additional Inherited Members

6.18.1 Detailed Description

Clase para resolucil sistema lineal mediante Gauss-Seidel.
Author

Antonio Carrillo Ledesma

Date
primavera 2009
Version
1.0.1
Bug No hay errores conocidos

6.18.2 Constructor & Destructor Documentation

6.18.2.1 ResuelveGaussSeidel::ResuelveGaussSeidel (void) [inline]

Constructor de la clase.

6.18.2.2 ResuelveGaussSeidel::ResuelveGaussSeidel (Vector * x, Vector * b) [inline]

Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.18.2.3 ResuelveGaussSeidel::ResuelveGaussSeidel (Vector * x, Vector * b, Idouble ep, int iter) [inline]

Constructor de la clase Constructor de la clase

Parameters

| | |
|-------------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>ep</i> | Tolerancia del mdo |
| <i>iter</i> | Nmero mmo de iteraciones |

6.18.3 Member Data Documentation

6.18.3.1 Idouble ResuelveGaussSeidel::Ep [protected]

Tolerancia.

6.18.3.2 int ResuelveGaussSeidel::Iter [protected]

Iteraciones mmas.

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

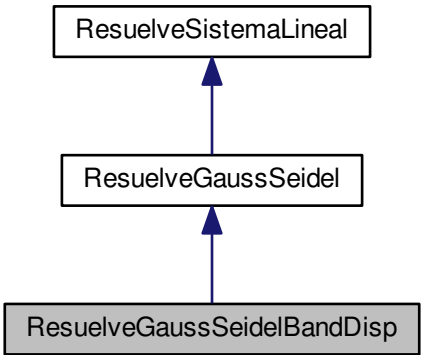
- [ResuelveGaussSeidel.hpp](#)

6.19 ResuelveGaussSeidelBandDisp Class Reference

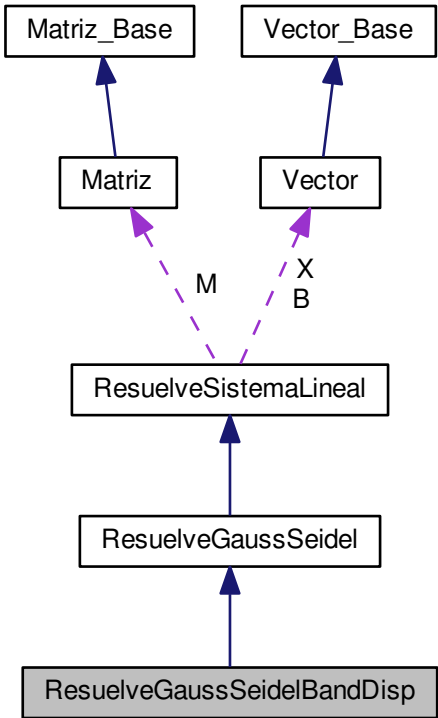
Clase para resolucil sistema lineal mediante Gauss-Seidel.

#include <ResuelveGaussSeidelBandDisp.hpp>

Inheritance diagram for ResuelveGaussSeidelBandDisp:



Collaboration diagram for ResuelveGaussSeidelBandDisp:



Public Member Functions

- [ResuelveGaussSeidelBandDisp](#) (void)
Constructor de la clase.
- [ResuelveGaussSeidelBandDisp](#) ([MatrizBandDisp](#) *A)
Constructor de la clase.
- [ResuelveGaussSeidelBandDisp](#) ([MatrizBandDisp](#) *A, [Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- [ResuelveGaussSeidelBandDisp](#) ([MatrizBandDisp](#) *A, [Vector](#) *x, [Vector](#) *b, [ldouble](#) ep, int iter)
Constructor de la clase.
- void [resuelve](#) (void)
Resuelve el sistema lineal.
- void [resuelve](#) ([Vector](#) *x, [Vector](#) *b)
Resuelve el sistema lineal.

Additional Inherited Members

6.19.1 Detailed Description

Clase para resolucil sistema lineal mediante Gauss-Seidel.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.19.2 Constructor & Destructor Documentation

6.19.2.1 `ResuelveGaussSeidelBandDisp::ResuelveGaussSeidelBandDisp (void) [inline]`
Constructor de la clase.

6.19.2.2 `ResuelveGaussSeidelBandDisp::ResuelveGaussSeidelBandDisp (MatrizBandDisp * A) [inline]`
Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
|----------|--|

6.19.2.3 `ResuelveGaussSeidelBandDisp::ResuelveGaussSeidelBandDisp (MatrizBandDisp * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.19.2.4 `ResuelveGaussSeidelBandDisp::ResuelveGaussSeidelBandDisp (MatrizBandDisp * A, Vector * x, Vector * b, Idouble ep, int iter)` `[inline]`

Constructor de la clase.
Parameters

| | |
|-------------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>ep</i> | Tolerancia del mdo |
| <i>iter</i> | Nmero mmo de iteraciones |

6.19.3 Member Function Documentation

6.19.3.1 `void ResuelveGaussSeidelBandDisp::resuelve (void)` `[virtual]`

Resuelve el sistema lineal.
Implements [ResuelveSistemaLineal](#).

6.19.3.2 `void ResuelveGaussSeidelBandDisp::resuelve (Vector * x, Vector * b)` `[inline]`,
`[virtual]`

Resuelve el sistema lineal.
Parameters

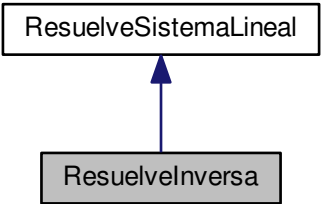
| | |
|----------|--|
| <i>x</i> | Puntero a un vector, solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

Implements [ResuelveSistemaLineal](#).
The documentation for this class was generated from the following files:

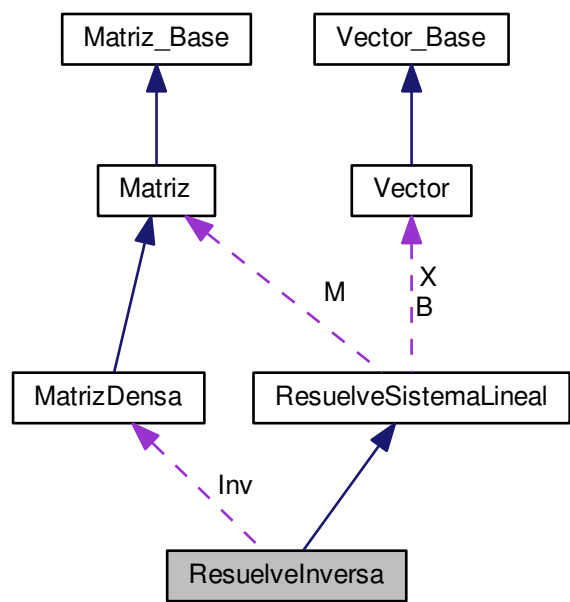
- [ResuelveGaussSeidelBandDisp.hpp](#)
- [ResuelveGaussSeidelBandDisp.cpp](#)

6.20 ResuelveInversa Class Reference

Clase para resolucil sistema lineal mediante el uso de la matriz inversa.
`#include <ResuelveInversa.hpp>`
Inheritance diagram for ResuelveInversa:



Collaboration diagram for ResuelveInversa:



Public Member Functions

- [ResuelveInversa](#) (void)
Constructor de la clase.
- [ResuelveInversa](#) (Matriz *A)
Constructor de la clase.
- [ResuelveInversa](#) (Matriz *A, Vector *x, Vector *b)
Constructor de la clase.
- [~ResuelveInversa](#) (void)
Destructor de la clase.
- void [resuelve](#) (void)
Resuelve el sistema lineal.
- void [resuelve](#) (Vector *x, Vector *b)
Resuelve el sistema lineal.
- void [invierte](#) (Matriz *A, Matriz *inv)
Calcula la inversa de una matriz usando el mdo de eliminaciussiana.

Protected Attributes

- bool [MatrizInvertida](#)
Matriz factorizada.
- [MatrizDensa](#) * [Inv](#)

Additional Inherited Members

6.20.1 Detailed Description

Clase para resolucil sistema lineal mediante el uso de la matriz inversa.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.20.2 Constructor & Destructor Documentation

6.20.2.1 [ResuelveInversa::ResuelveInversa \(void \)](#) [inline]
Constructor de la clase.

6.20.2.2 [ResuelveInversa::ResuelveInversa \(Matriz * A \)](#) [inline]
Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo Matriz |
|----------|--|

6.20.2.3 `ResuelveInversa::ResuelveInversa (Matriz * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo Matriz |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.20.2.4 `ResuelveInversa::~~ResuelveInversa (void)` `[inline]`

Destructor de la clase.

6.20.3 Member Function Documentation

6.20.3.1 `void ResuelveInversa::invierte (Matriz * A, Matriz * inv)`

Calcula la inversa de una matriz usando el mdo de eliminaciussiana.

Parameters

| | |
|------------|---|
| <i>A</i> | Puntero a una matriz tipo Matriz |
| <i>inv</i> | Puntero a una matriz tipo Matriz que contendra la inversa |

6.20.3.2 `void ResuelveInversa::resuelve (void)` `[inline],[virtual]`

Resuelve el sistema lineal.

Implements [ResuelveSistemaLineal](#).

6.20.3.3 `void ResuelveInversa::resuelve (Vector * x, Vector * b)` `[inline],[virtual]`

Resuelve el sistema lineal.

Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un vector, solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

Implements [ResuelveSistemaLineal](#).

6.20.4 Member Data Documentation

6.20.4.1 `MatrizDensa* ResuelveInversa::Inv` `[protected]`

6.20.4.2 `bool ResuelveInversa::MatrizInvertida` `[protected]`

[Matriz](#) factorizada.

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

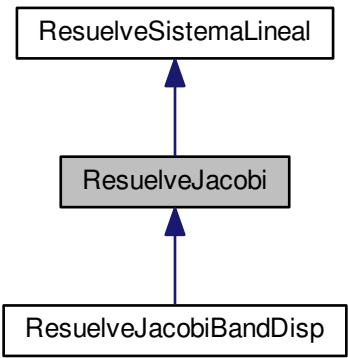
- [ResuelveInversa.hpp](#)
- [ResuelveInversa.cpp](#)

6.21 ResuelveJacobi Class Reference

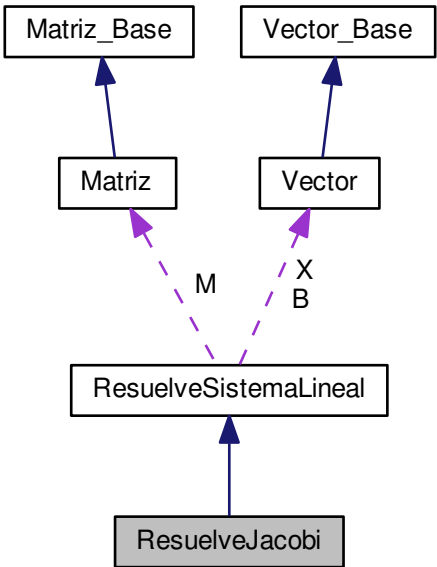
Clase para resolucil sistema lineal mediante Jacobi.

`#include <ResuelveJacobi.hpp>`

Inheritance diagram for ResuelveJacobi:



Collaboration diagram for ResuelveJacobi:



Public Member Functions

- [ResuelveJacobi](#) (void)
Constructor de la clase.
- [ResuelveJacobi](#) ([Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- [ResuelveJacobi](#) ([Vector](#) *x, [Vector](#) *b, [ldouble](#) ep, int iter)
Constructor de la clase.

Protected Attributes

- [ldouble](#) Ep
Tolerancia.
- int [Iter](#)
Iteraciones mmas.

Additional Inherited Members

6.21.1 Detailed Description

Clase para resolucil sistema lineal mediante Jacobi.
Author

Antonio Carrillo Ledesma

Date
primavera 2009
Version
1.0.1
Bug No hay errores conocidos

6.21.2 Constructor & Destructor Documentation

6.21.2.1 ResuelveJacobi::ResuelveJacobi (void) [inline]

Constructor de la clase.

6.21.2.2 ResuelveJacobi::ResuelveJacobi (Vector * x, Vector * b) [inline]

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.21.2.3 ResuelveJacobi::ResuelveJacobi (Vector * x, Vector * b, Idouble ep, int iter) [inline]

Constructor de la clase.
Parameters

| | |
|-------------|--|
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>ep</i> | Tolerancia del mdo |
| <i>iter</i> | Nmero mmo de iteraciones |

6.21.3 Member Data Documentation

6.21.3.1 Idouble ResuelveJacobi::Ep [protected]

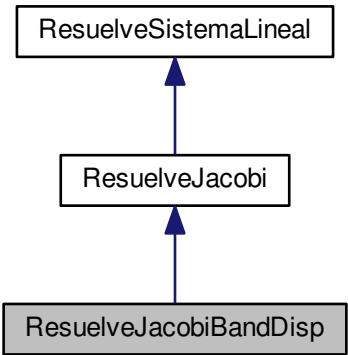
Tolerancia.

6.21.3.2 int ResuelveJacobi::Iter [protected]

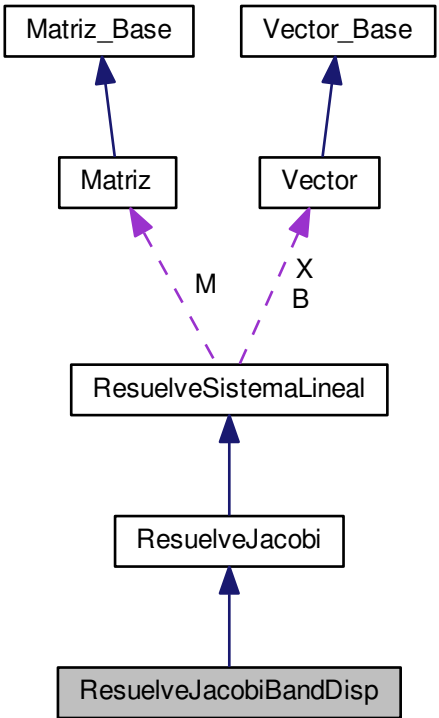
Iteraciones mmas.
The documentation for this class was generated from the following file:
· [ResuelveJacobi.hpp](#)

6.22 ResuelveJacobiBandDisp Class Reference

Clase para resolucil sistema lineal mediante Jacobi.
#include <ResuelveJacobiBandDisp.hpp>
Inheritance diagram for ResuelveJacobiBandDisp:



Collaboration diagram for ResuelveJacobiBandDisp:



Public Member Functions

- [ResuelveJacobiBandDisp](#) (void)
Constructor de la clase.
- [ResuelveJacobiBandDisp](#) ([MatrizBandDisp](#) *A)
Constructor de la clase.
- [ResuelveJacobiBandDisp](#) ([MatrizBandDisp](#) *A, [Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- void [resuelve](#) (void)
Resuelve el sistema lineal.
- void [resuelve](#) ([Vector](#) *x, [Vector](#) *b)
Resuelve el sistema lineal.

Additional Inherited Members

6.22.1 Detailed Description

Clase para resolucil sistema lineal mediante Jacobi.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.22.2 Constructor & Destructor Documentation

6.22.2.1 [ResuelveJacobiBandDisp::ResuelveJacobiBandDisp](#) (void) [inline]

Constructor de la clase.

6.22.2.2 [ResuelveJacobiBandDisp::ResuelveJacobiBandDisp](#) ([MatrizBandDisp](#) * A) [inline]

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
|----------|--|

6.22.2.3 `ResuelveJacobiBandDisp::ResuelveJacobiBandDisp (MatrizBandDisp * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.
Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz del tipo MatrizBandDisp |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.22.3 Member Function Documentation

6.22.3.1 `void ResuelveJacobiBandDisp::resuelve (void)` `[virtual]`

Resuelve el sistema lineal.
Implements [ResuelveSistemaLineal](#).

6.22.3.2 `void ResuelveJacobiBandDisp::resuelve (Vector * x, Vector * b)` `[inline],[virtual]`

Resuelve el sistema lineal.
Parameters

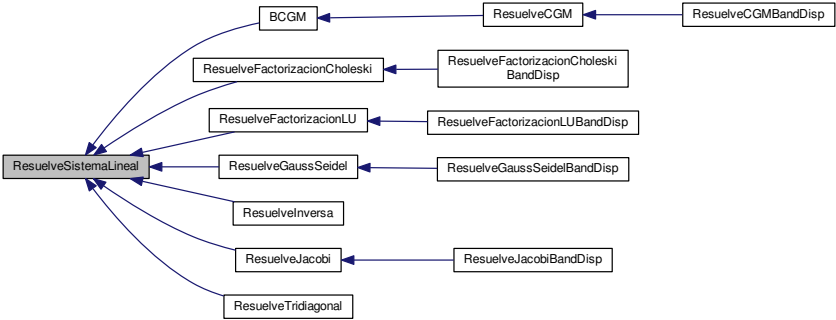
| | |
|----------|--|
| <i>x</i> | Puntero a un vector, solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

Implements [ResuelveSistemaLineal](#).
The documentation for this class was generated from the following files:

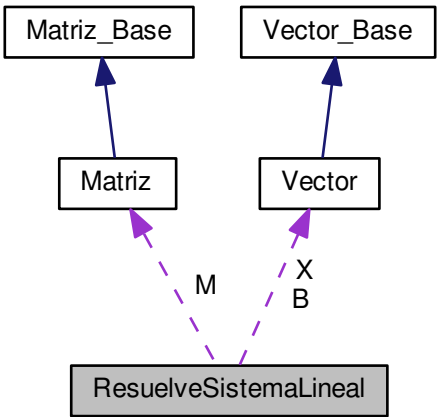
- [ResuelveJacobiBandDisp.hpp](#)
- [ResuelveJacobiBandDisp.cpp](#)

6.23 ResuelveSistemaLineal Class Reference

Clase para resolucil sistema lineal.
`#include <ResuelveSistemaLineal.hpp>`
Inheritance diagram for ResuelveSistemaLineal:



Collaboration diagram for ResuelveSistemaLineal:



Public Member Functions

- void **inicializa** (Matriz *m, Vector *b, Vector *x)
Resuelve el sistema lineal.
- **ResuelveSistemaLineal** (void)
Constructor de la clase.
- virtual void **resuelve** (void)=0
Resuelve el sistema lineal.
- virtual void **resuelve** (Vector *b, Vector *x)=0
Resuelve el sistema lineal.
- int **retornaNumeroliteraciones** (void)
Retorna el nmero de iteraciones en la resolucil sistema lineal.
- bool **elMetodoModificaMatriz** (void)
Indica si el mdo numco modifica la matriz.
- int **retornaMetodoNumerico** (void)
Retorna el mdo numco.
- int **retornaRequiereMatriz** (void)
Retorna el tipo de matriz que requiere el mdo.
- void **informacionMetodo** (void)
Visualiza la informacil mdo.

Protected Member Functions

- void **error** (const char *txt)
Visualiza el error indicado y termina el programa.

Protected Attributes

- bool **MetodoModificaMatriz**
Indica si el mdo numco modifica la matriz.
- int **NumIteraciones**
Numero de iteraciones en la resolucil sistema lineal.
- int **MetodoNumerico**
Mdo Numerico.
- int **RequiereMatriz**
Requiere matriz del tipo Bandada, Dispersa, Densa.
- Vector * **X**
Vector soluci
- Vector * **B**
Vector lado derecho.
- Matriz * **M**
Matriz.

6.23.1 Detailed Description

Clase para resolucil sistema lineal.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.23.2 Constructor & Destructor Documentation

6.23.2.1 `ResuelveSistemaLineal::ResuelveSistemaLineal (void)` `[inline]`

Constructor de la clase.

6.23.3 Member Function Documentation

6.23.3.1 `bool ResuelveSistemaLineal::elMetodoModificaMatriz (void)` `[inline]`

Indica si el mdo numco modifica la matriz.
Returns

(true) si el mdo numco modifica la matriz, (false) en caso contrario

6.23.3.2 `void ResuelveSistemaLineal::error (const char * txt)` `[inline]`, `[protected]`

Visualiza el error indicado y termina el programa.
Parameters

| | |
|------------|-----------------------------|
| <i>txt</i> | Puntero al mensaje de error |
|------------|-----------------------------|

6.23.3.3 `void ResuelveSistemaLineal::informacionMetodo (void)` `[inline]`

Visualiza la informacil mdo.

6.23.3.4 `void ResuelveSistemaLineal::inicializa (Matriz * m, Vector * b, Vector * x)` `[inline]`

Resuelve el sistema lineal.
Parameters

| | |
|----------|--|
| <i>m</i> | Puntero a la matriz del sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>x</i> | Puntero a un vector, solucil sistema lineal |

6.23.3.5 `virtual void ResuelveSistemaLineal::resuelve (void)` `[pure virtual]`

Resuelve el sistema lineal.
Implemented in [ResuelveFactorizacionCholeskiBandDisp](#), [ResuelveFactorizacionLUBandDisp](#), [ResuelveInversa](#), [ResuelveTridiagonal](#), [ResuelveGaussSeidelBandDisp](#), [BCGM](#), and [ResuelveJacobiBandDisp](#).

6.23.3.6 `virtual void ResuelveSistemaLineal::resuelve (Vector * b, Vector * x)` `[pure virtual]`

Resuelve el sistema lineal.
Parameters

| | |
|----------|--|
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |
| <i>x</i> | Puntero a un vector, solucil sistema lineal |

Implemented in [ResuelveInversa](#), [ResuelveFactorizacionCholeskiBandDisp](#), [ResuelveFactorizacionLUBandDisp](#), [ResuelveTridiagonal](#), [ResuelveGaussSeidelBandDisp](#), [BCGM](#), and [ResuelveJacobiBandDisp](#).

6.23.3.7 `int ResuelveSistemaLineal::retornaMetodoNumerico (void)` `[inline]`

Retorna el mdo numco.
Returns
Retorna el mdo numco CGM, FACT_LU, GAUSSSEIDEL, INVERSA, JACOBI, TRIDIAGONAL, etc

6.23.3.8 `int ResuelveSistemaLineal::retornaNumeroliteraciones (void)` `[inline]`

Retorna el nmero de iteraciones en la resolucil sistema lineal.
Returns
Nmero de iteraciones

6.23.3.9 `int ResuelveSistemaLineal::retornaRequiereMatriz (void)` `[inline]`

Retorna el tipo de matriz que requiere el mdo.
Returns
Retorna el tipo de matriz Bandada, Dispersa, Densa

6.23.4 Member Data Documentation

6.23.4.1 `Vector* ResuelveSistemaLineal::B` [protected]

Vector lado derecho.

6.23.4.2 `Matriz* ResuelveSistemaLineal::M` [protected]

[Matriz](#).

6.23.4.3 `bool ResuelveSistemaLineal::MetodoModificaMatriz` [protected]

Indica si el mdo numco modifica la matriz.

6.23.4.4 `int ResuelveSistemaLineal::MetodoNumerico` [protected]

Mdo Numerico.

6.23.4.5 `int ResuelveSistemaLineal::NumIteraciones` [protected]

Numero de iteraciones en la resolucil sistema lineal.

6.23.4.6 `int ResuelveSistemaLineal::RequiereMatriz` [protected]

Requiere matriz del tipo Bandada, Dispersa, Densa.

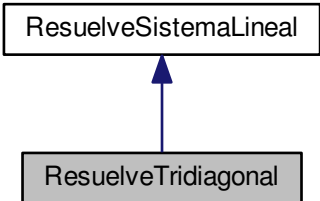
6.23.4.7 `Vector* ResuelveSistemaLineal::X` [protected]

[Vector](#) soluci

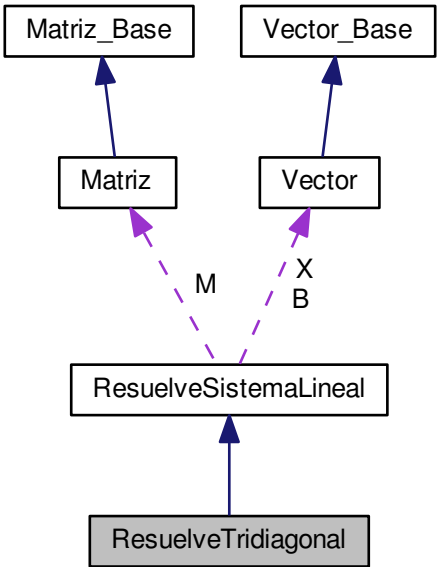
The documentation for this class was generated from the following file:
· [ResuelveSistemaLineal.hpp](#)

6.24 ResuelveTridiagonal Class Reference

Clase para resolucil sistema lineal tridiagonal.
`#include <ResuelveTridiagonal.hpp>`
Inheritance diagram for ResuelveTridiagonal:



Collaboration diagram for ResuelveTridiagonal:



Public Member Functions

- [ResuelveTridiagonal](#) (void)
Constructor de la clase.
- [ResuelveTridiagonal](#) ([Matriz](#) *A)
Constructor de la clase.
- [ResuelveTridiagonal](#) ([Matriz](#) *A, [Vector](#) *x, [Vector](#) *b)
Constructor de la clase.
- void [resuelve](#) (void)
Resuelve el sistema lineal.
- void [resuelve](#) ([Vector](#) *x, [Vector](#) *b)
Resuelve el sistema lineal.

Additional Inherited Members

6.24.1 Detailed Description

Clase para resolucil sistema lineal tridiagonal.

Author

Antonio Carrillo Ledesma

Date

primavera 2009

Version

1.0.1

Bug No hay errores conocidos

6.24.2 Constructor & Destructor Documentation

6.24.2.1 `ResuelveTridiagonal::ResuelveTridiagonal (void)` `[inline]`

Constructor de la clase.

6.24.2.2 `ResuelveTridiagonal::ResuelveTridiagonal (Matriz * A)` `[inline]`

Constructor de la clase.

Parameters

| | |
|---|--|
| A | Puntero a una matriz tipo Matriz |
|---|--|

6.24.2.3 `ResuelveTridiagonal::ResuelveTridiagonal (Matriz * A, Vector * x, Vector * b)` `[inline]`

Constructor de la clase.

Parameters

| | |
|----------|--|
| <i>A</i> | Puntero a una matriz tipo Matriz |
| <i>x</i> | Puntero a un Vector , solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

6.24.3 Member Function Documentation

6.24.3.1 void ResuelveTridiagonal::resuelve (void) [virtual]

Resuelve el sistema lineal.
Implements [ResuelveSistemaLineal](#).

6.24.3.2 void ResuelveTridiagonal::resuelve (Vector * x, Vector * b) [inline],[virtual]

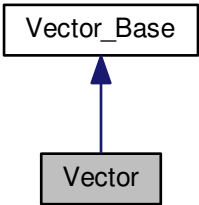
Resuelve el sistema lineal.
Parameters

| | |
|----------|--|
| <i>x</i> | Puntero a un vector, solucil sistema lineal |
| <i>b</i> | Puntero a un vector, lado derecho del sistema lineal |

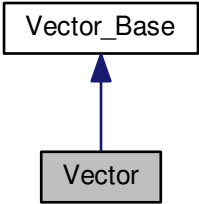
Implements [ResuelveSistemaLineal](#).
The documentation for this class was generated from the following files:
• [ResuelveTridiagonal.hpp](#)
• [ResuelveTridiagonal.cpp](#)

6.25 Vector Class Reference

Clase para el trabajar con vectores de punto flotante.
`#include <Vector.hpp>`
Inheritance diagram for Vector:



Collaboration diagram for Vector:



Public Member Functions

- [Vector](#) (void)
Constructor de la clase.
- [Vector](#) ([Vector](#) &vec)
Constructor de la clase.
- [Vector](#) (const int col)
Constructor de la clase.
- [Vector](#) (const int col, const [ldouble](#) val)
Constructor de la clase.
- [Vector](#) (const int col, const [ldouble](#) val, const char *nmb)
Constructor de la clase.
- [Vector](#) (const int col, const char *nmb)
Constructor de la clase.

- `~Vector ()`
Destructor de la clase.
- void `redimensiona (int ncol)`
Redimensiona el tamal vector.
- void `inicializa (const Idouble val)`
Inicializa el vector al valor val indicado.
- `Idouble operator[] (size_t col)`
Retorna el valor de la columna col.
- `Idouble operator() (size_t col)`
Retorna el valor de la columna col.
- void `asigna (const int col, const Idouble val)`
Asigna el valor val a la columna col.
- `Idouble retorna (const int col)`
Retorna el valor de la columna col.
- void `visualiza (const int tp, int nl=0, int num=0)`
Visualiza el vector.
- void `copia (Vector *a)`
Copia el contenido del vector al vector a.
- void `convierte (Idouble *a, int tam)`
Convierte el contenido del arreglo a de tamam a vector.
- void `suma (Vector *a, Vector *b)`
Suma los vectores a y b.
- void `suma (Vector *a)`
Suma al vector a.
- void `resta (Vector *a, Vector *b)`
Resta los vectores a menos b.
- void `resta (Vector *a)`
Resta el vector a.
- void `multiplica (Idouble esc)`
Multiplica el vector por el escalar esc.
- `Idouble normalInf (void)`
Calcula la norma infinito.
- `Idouble productoPunto (void)`
Calcula el producto punto.
- `Idouble productoPunto (Vector *a)`
Calcula el producto punto.
- int `esVectorCero (void)`
Es el vector cero.
- int `esCadaEntradaMasPequeno (double eps)`
Es el vector en todas sus componentes mequee eps.
- `Idouble tamano (void)`
Tamal vector (aproximado) en Kb.
- void `visualizaInformacion (void)`
Visualiza informacineral del vector.

Protected Member Functions

- void `solicitaMemoria (const int col, const Idouble val)`
Solicita la memoria necesaria para contener los valores del vector.
- void `liberaMemoria (int tp)`
Libera la memoria solicitada dinamicamente.

Protected Attributes

- `Idouble * V`
Puntero al contenido del Vector.

6.25.1 Detailed Description

Clase para el trabajar con vectores de punto flotante.
Author

Antonio Carrillo Ledesma
Date

primavera 2009
Version

1.0.1
Bug No hay errores conocidos

6.25.2 Constructor & Destructor Documentation

6.25.2.1 Vector::Vector (void) [inline]

Constructor de la clase.

6.25.2.2 Vector::Vector (Vector & vec) [inline]

Constructor de la clase.
Genera un Vector a partir de los datos de otro

Parameters

| | |
|------------|---|
| <i>vec</i> | Vector del cual se hara una copia |
|------------|---|

6.25.2.3 `Vector::Vector (const int col)` `[inline]`

Constructor de la clase.
Genera un [Vector](#) del tamapecificado por el nmero de columnas e inicializado por omisiero.
Parameters

| | |
|------------|------------------------------|
| <i>col</i> | Nmero de columnas del vector |
|------------|------------------------------|

6.25.2.4 `Vector::Vector (const int col, const ldouble val)` `[inline]`

Constructor de la clase.
Genera un [Vector](#) del tamapecificado por el nmero de columnas e inicializado por omisi valor val.
Parameters

| | |
|------------|---|
| <i>col</i> | Nmero de columnas del vector |
| <i>val</i> | Valor por omisira inicializar el vector |

6.25.2.5 `Vector::Vector (const int col, const ldouble val, const char * nmb)` `[inline]`

Constructor de la clase.
Genera un [Vector](#) del tamapecificado por el nmero de columnas e inicializado por omisi valor val.
Parameters

| | |
|------------|---|
| <i>col</i> | Nmero de columnas del vector |
| <i>val</i> | Valor por omisira inicializar el vector |
| <i>nmb</i> | Nombre del vector |

6.25.2.6 `Vector::Vector (const int col, const char * nmb)` `[inline]`

Constructor de la clase.
Genera un [Vector](#) del tamapecificado por el nmero de columnas e inicializado por omisiero.
Parameters

| | |
|------------|-------------------------------|
| <i>col</i> | Numero de columnas del vector |
| <i>nmb</i> | Nombre del vector |

6.25.2.7 `Vector::~~Vector ()` `[inline]`

Destructor de la clase.

6.25.3 Member Function Documentation

6.25.3.1 `void Vector::asigna (const int col, const ldouble val)` `[inline]`

Asigna el valor val a la columna col.
Parameters

| | |
|------------|---------|
| <i>col</i> | Columna |
| <i>val</i> | Valor |

6.25.3.2 `void Vector::convierte (ldouble * a, int tam)`

Convierte el contenido del arreglo a de tamam a vector.
Parameters

| | |
|------------|--------------------------------|
| <i>a</i> | Arreglo de tipo <i>ldouble</i> |
| <i>tam</i> | Tamal arreglo |

6.25.3.3 `void Vector::copia (Vector * a)`

Copia el contenido del vector al vector a.
Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a un vector |
|----------|---------------------|

6.25.3.4 **int** Vector::esCadaEntradaMasPequeno (**double** *eps*)

Es el vector en todas sus componentes mequee eps.

Parameters

| | |
|------------|------------|
| <i>eps</i> | Tolerancia |
|------------|------------|

Returns

(1) si todas las componentes son mequee eps, (0) en caso contrario

6.25.3.5 **int** Vector::esVectorCero (**void**)

Es el vector cero.

Returns

(1) si es el vector cero, (0)en caso contrario

6.25.3.6 **void** Vector::inicializa (**const** **Idouble** *val*)

Inicializa el vector al valor val indicado.

Parameters

| | |
|------------|---|
| <i>val</i> | Valor por omisira inicializar el vector |
|------------|---|

6.25.3.7 **void** Vector::liberaMemoria (**int** *tp*) *[inline], [protected]*

Libera la memoria solicitada dinamicamente.

Parameters

| | |
|-----------|---|
| <i>tp</i> | Distinto de cero libera toda la memoria |
|-----------|---|

6.25.3.8 **void** Vector::multiplica (**Idouble** *esc*)

Multiplica el vector por el escalar esc.

Parameters

| | |
|------------|---------------|
| <i>esc</i> | Valor escalar |
|------------|---------------|

6.25.3.9 **Idouble** Vector::normalnf (**void**)

Calcula la norma infinito.

Returns

Valor del la norma infinita

6.25.3.10 **Idouble** Vector::operator() (**size_t** *col*) *[inline]*

Retorna el valor de la columna col.

Parameters

| | |
|------------|---------|
| <i>col</i> | Columna |
|------------|---------|

Returns

Valor

6.25.3.11 **Idouble** Vector::operator[] (**size_t** *col*) *[inline]*

Retorna el valor de la columna col.

Parameters

| | |
|------------|---------|
| <i>col</i> | Columna |
|------------|---------|

Returns

Valor

6.25.3.12 **Idouble** Vector::productoPunto (**void**)

Calcula el producto punto.

Returns

Valor del producto punto

6.25.3.13 **Idouble** Vector::productoPunto (**Vector** * *a*)

Calcula el producto punto.

Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a un vector |
|----------|---------------------|

Returns

Valor del producto punto

6.25.3.14 void Vector::redimensiona (int *ncol*) [inline]

Redimensiona el tamal vector.

Parameters

| | |
|-------------|-------------------------|
| <i>ncol</i> | Nuevo nmero de columnas |
|-------------|-------------------------|

6.25.3.15 void Vector::resta (Vector * *a*, Vector * *b*)

Resta los vectores a menos b.

Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a un vector |
| <i>b</i> | Puntero a un vector |

6.25.3.16 void Vector::resta (Vector * *a*)

Resta el vector a.

Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a un vector |
|----------|---------------------|

6.25.3.17 Idouble Vector::retorna (const int *col*) [inline]

Retorna el valor de la columna col.

Parameters

| | |
|------------|---------|
| <i>col</i> | Columna |
|------------|---------|

Returns

Valor

6.25.3.18 void Vector::solicitaMemoria (const int *col*, const Idouble *val*) [protected]

Solicita la memoria necesaria para contener los valores del vector.

Parameters

| | |
|------------|--|
| <i>col</i> | Nmero de columnas del vector |
| <i>val</i> | Valor por omision para inicializar el vector |

6.25.3.19 void Vector::suma (Vector * *a*, Vector * *b*)

Suma los vectores a y b.

Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a un vector |
| <i>b</i> | Puntero a un vector |

6.25.3.20 void Vector::suma (Vector * *a*)

Suma al vector a.

Parameters

| | |
|----------|---------------------|
| <i>a</i> | Puntero a un vector |
|----------|---------------------|

6.25.3.21 Idouble Vector::tamano (void) [inline]

Tamal vector (aproximado) en Kb.

Returns

Tamal vector en Kb

6.25.3.22 void Vector::visualiza (const int *tp*, int *nl* = 0, int *num* = 0)

Visualiza el vector.

Parameters

| | |
|------------|--|
| <i>tp</i> | (1) Se visualiza el vector de en formato de notacientca, (0) formato notaci punto flotante |
| <i>nl</i> | (1) Visualiza el vector en forma renglones, (0) en forma de columnas |
| <i>num</i> | (1) Numera las componentes del vector, (0) no lo hace |

6.25.3.23 void Vector::visualizaInformacion (void) [inline],[virtual]

Visualiza informacineral del vector.
Implements [Vector_Base](#).

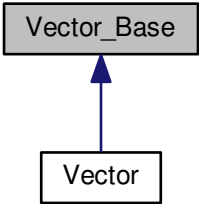
6.25.4 Member Data Documentation

6.25.4.1 Idouble* Vector::V [protected]

Puntero al contenido del [Vector](#).
The documentation for this class was generated from the following files:
· [Vector.hpp](#)
· [Vector.cpp](#)

6.26 Vector_Base Class Reference

Clase base para el trabajar con vectores.
`#include <Vector_Base.hpp>`
Inheritance diagram for Vector_Base:



Public Member Functions

- [Vector_Base](#) (void)
Constructor de la clase.
- [~Vector_Base](#) ()
Destructor de la clase.
- void [asignaNombre](#) (const char *nmb)
Asigna nombre al vector.
- char * [nombre](#) (void)
Retorna el nombre del vector.
- int [columnas](#) (void)
Retorna el nmero de columnas del vector.
- virtual void [visualizaInformacion](#) (void)=0
Visualiza informacineral del vector.

Protected Attributes

- int [Col](#)
Nmero de columnas del vector.
- char * [Nmb](#)
Nombre del vector.

6.26.1 Detailed Description

Clase base para el trabajar con vectores.
Author
Antonio Carrillo Ledesma
Date
primavera 2009
Version
1.0.1
[Bug](#) No hay errores conocidos

6.26.2 Constructor & Destructor Documentation

6.26.2.1 Vector_Base::Vector_Base (void) [inline]

Constructor de la clase.

6.26.2.2 Vector_Base::~~Vector_Base () [inline]

Destructor de la clase.

6.26.3 Member Function Documentation

6.26.3.1 void Vector_Base::asignaNombre (const char * nmb) [inline]

Asigna nombre al vector.

Parameters

| | |
|-----|-------------------|
| nmb | Nombre del vector |
|-----|-------------------|

6.26.3.2 int Vector_Base::columnas (void) [inline]

Retorna el nmero de columnas del vector.
Returns

Regresa el nmero de columnas del vector

6.26.3.3 char* Vector_Base::nombre (void) [inline]

Retorna el nombre del vector.
Returns

Regresa un puntero al nombre del vector

6.26.3.4 virtual void Vector_Base::visualizaInformacion (void) [pure virtual]

Visualiza informacin del vector.
Implemented in [Vector](#).

6.26.4 Member Data Documentation

6.26.4.1 int Vector_Base::Col [protected]

Nmero de columnas del vector.

6.26.4.2 char* Vector_Base::Nmb [protected]

Nombre del vector.

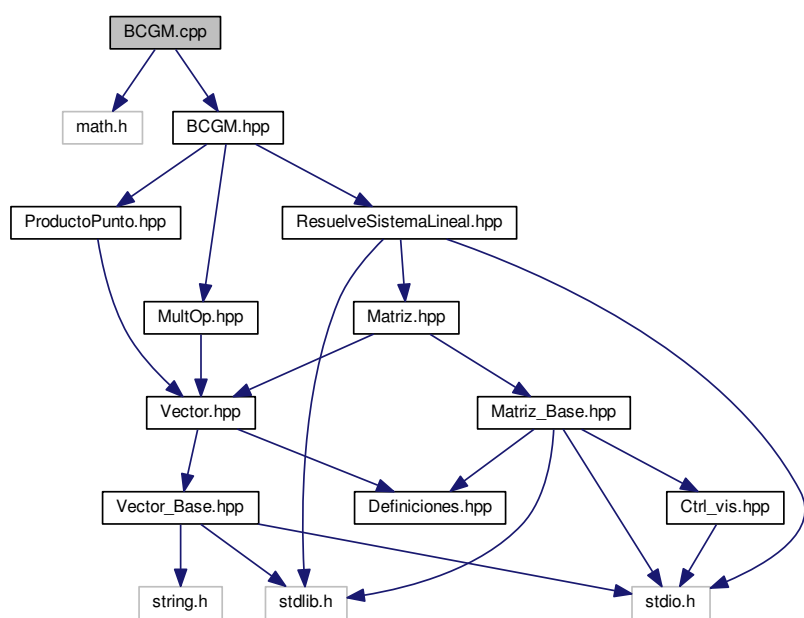
The documentation for this class was generated from the following file:
· [Vector_Base.hpp](#)

Chapter 7

File Documentation

7.1 BCGM.cpp File Reference

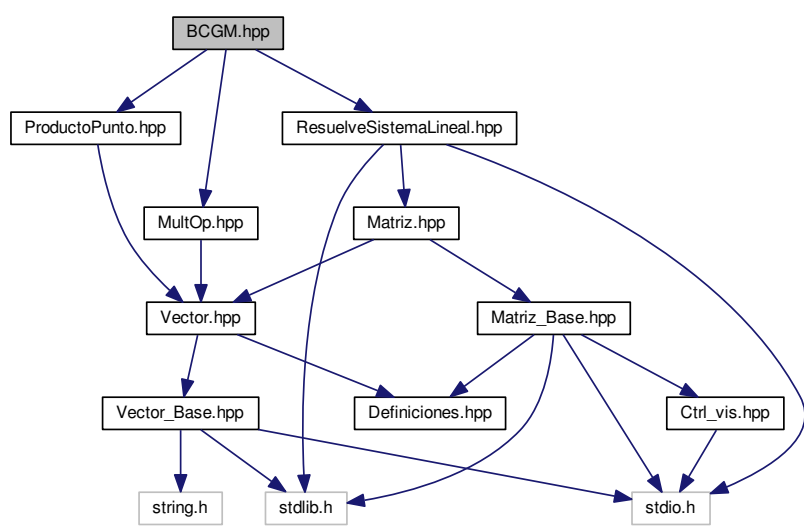
```
#include <math.h>
#include "BCGM.hpp"
Include dependency graph for BCGM.cpp:
```



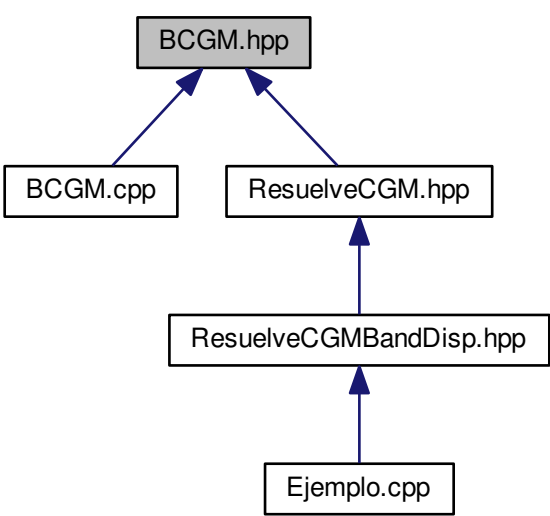
7.2 BCGM.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
#include "MultOp.hpp"
#include "ProductoPunto.hpp"
```

Include dependency graph for BCGM.hpp:



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

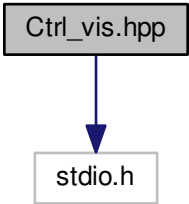


Classes

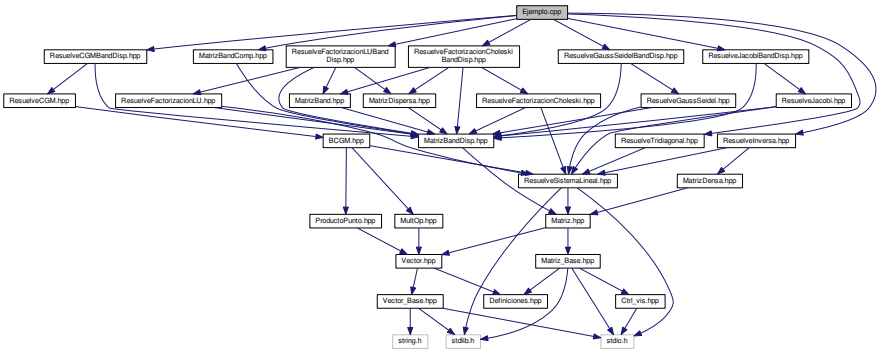
- class [BCGM](#)
Clase para resolucil sistema lineal mediante CGM.

7.3 Ctrl_vis.hpp File Reference

#include <stdio.h>
Include dependency graph for Ctrl_vis.hpp:



Include dependency graph for Ejemplo.cpp:



Functions

- void Ejem1 (void)
- void Ejem2 (void)
- int main (void)

7.5.1 Function Documentation

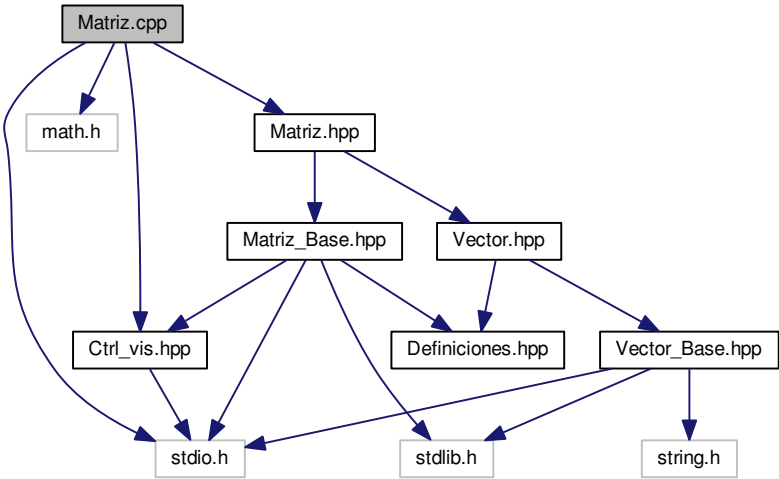
7.5.1.1 void Ejem1 (void)

7.5.1.2 void Ejem2 (void)

7.5.1.3 int main (void)

7.6 Matriz.cpp File Reference

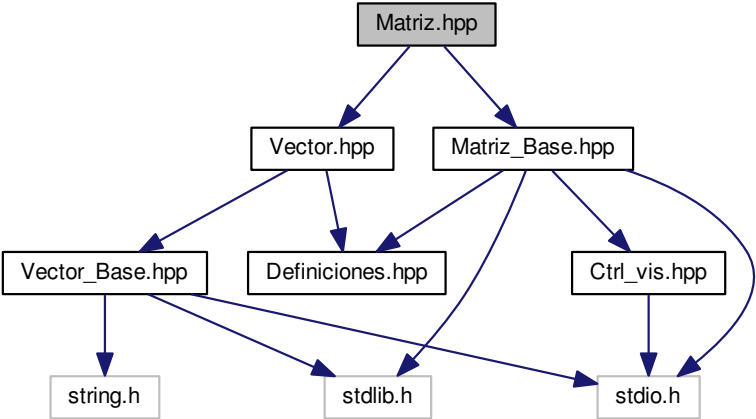
```
#include <stdio.h>
#include <math.h>
#include "Matriz.hpp"
#include "Ctrl_vis.hpp"
Include dependency graph for Matriz.cpp:
```



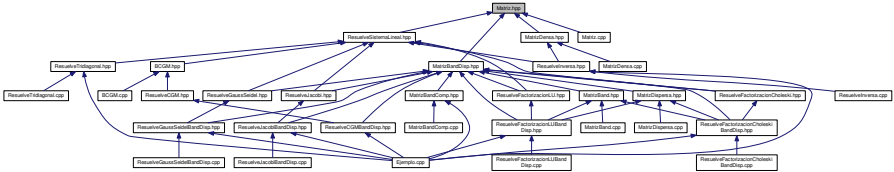
7.7 Matriz.hpp File Reference

```
#include "Matriz_Base.hpp"
#include "Vector.hpp"
```


Include dependency graph for Matriz.hpp:



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

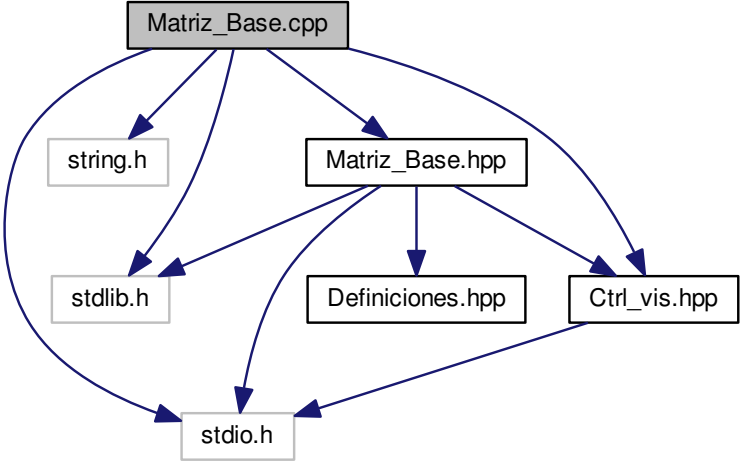


Classes

- class [Matriz](#)
Clase base para el trabajar con matrices de punto flotante.

7.8 Matriz_Base.cpp File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include "Matriz_Base.hpp"
#include "Ctrl_vis.hpp"
Include dependency graph for Matriz_Base.cpp:
```

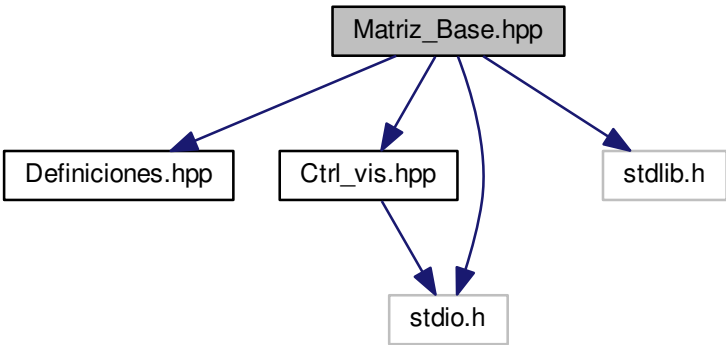


7.9 Matriz_Base.hpp File Reference

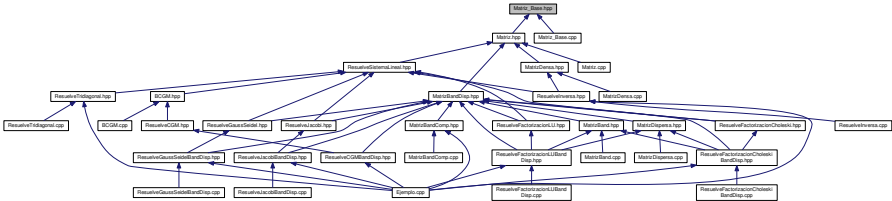
```
#include "Definiciones.hpp"
```

```
#include "Ctrl_vis.hpp"
#include <stdio.h>
#include <stdlib.h>
```

Include dependency graph for Matriz_Base.hpp:



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



Classes

- class [Matriz_Base](#)
Clase base para el trabajar con matrices.

Macros

- #define [MATRIZ_BANDADA](#) 0
- #define [MATRIZ_DENSA](#) 1
- #define [MATRIZ_ENTERA_DENSA](#) 2
- #define [MATRIZ_DISPERSA](#) 3
- #define [MATRIZ_BANDADA_COMPACTA](#) 4
- #define [MATRIZ_DISPERSA_COMPACTA](#) 5
- #define [MATRIZ_TAMANO_VARIABLE](#) 6
- #define [MATRIZ_RALA](#) 7

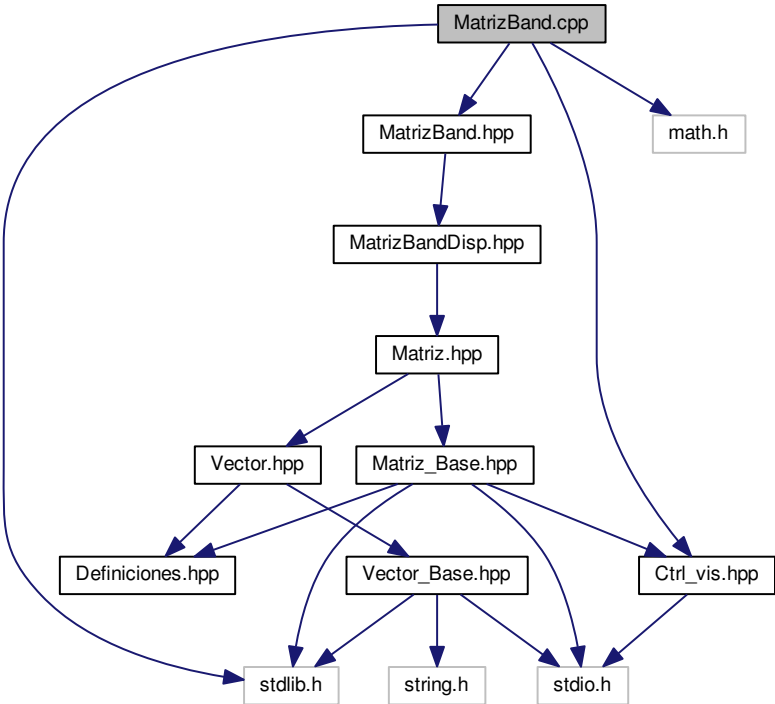
7.9.1 Macro Definition Documentation

- 7.9.1.1 #define [MATRIZ_BANDADA](#) 0
- 7.9.1.2 #define [MATRIZ_BANDADA_COMPACTA](#) 4
- 7.9.1.3 #define [MATRIZ_DENSA](#) 1
- 7.9.1.4 #define [MATRIZ_DISPERSA](#) 3
- 7.9.1.5 #define [MATRIZ_DISPERSA_COMPACTA](#) 5
- 7.9.1.6 #define [MATRIZ_ENTERA_DENSA](#) 2
- 7.9.1.7 #define [MATRIZ_RALA](#) 7
- 7.9.1.8 #define [MATRIZ_TAMANO_VARIABLE](#) 6

7.10 MatrizBand.cpp File Reference

```
#include <stdlib.h>
#include <math.h>
#include "MatrizBand.hpp"
#include "Ctrl_vis.hpp"
```

Include dependency graph for MatrizBand.cpp:



Macros

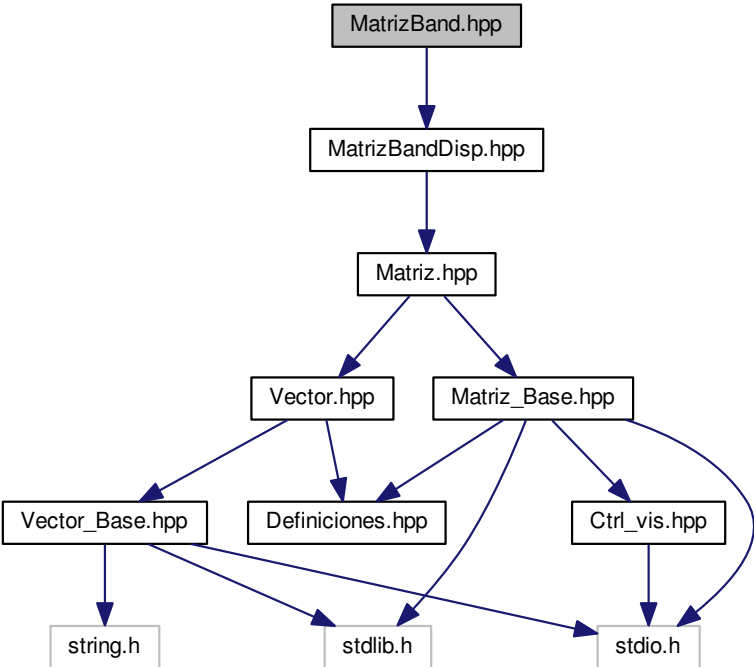
· #define EPS 1e-14

7.10.1 Macro Definition Documentation

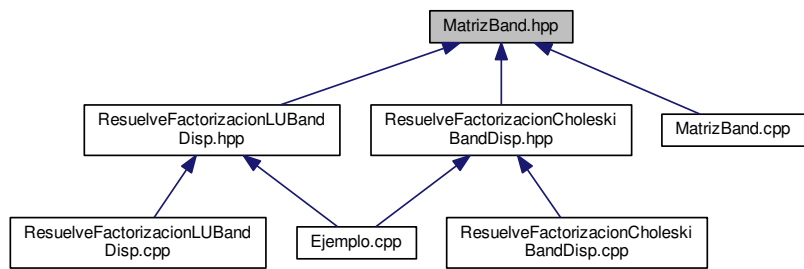
7.10.1.1 #define EPS 1e-14

7.11 MatrizBand.hpp File Reference

#include "MatrizBandDisp.hpp"
Include dependency graph for MatrizBand.hpp:



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

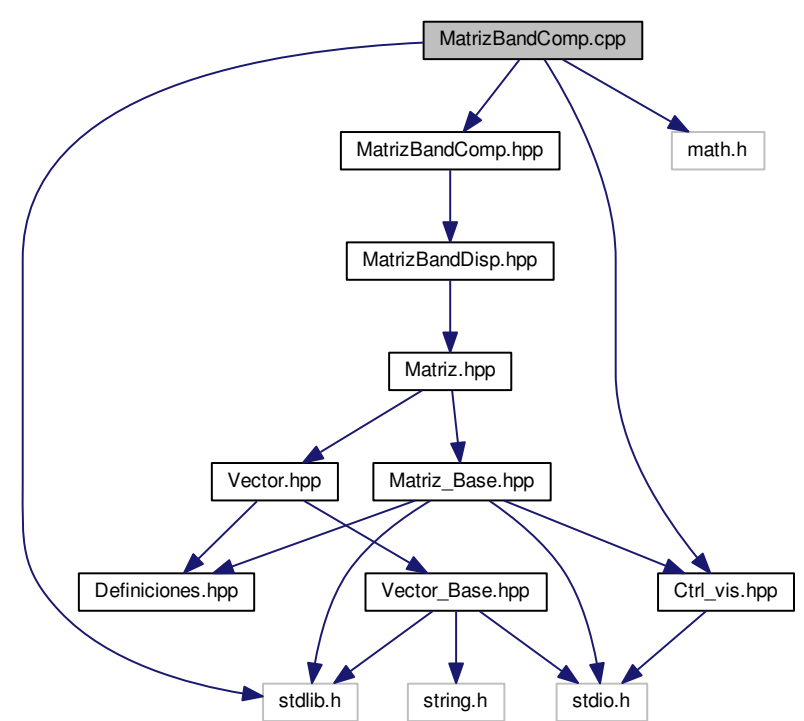


Classes

- class [MatrizBand](#)

7.12 MatrizBandComp.cpp File Reference

```
#include <stdlib.h>
#include <math.h>
#include "MatrizBandComp.hpp"
#include "Ctrl_vis.hpp"
Include dependency graph for MatrizBandComp.cpp:
```



Macros

- #define [EPS](#) 1e-14

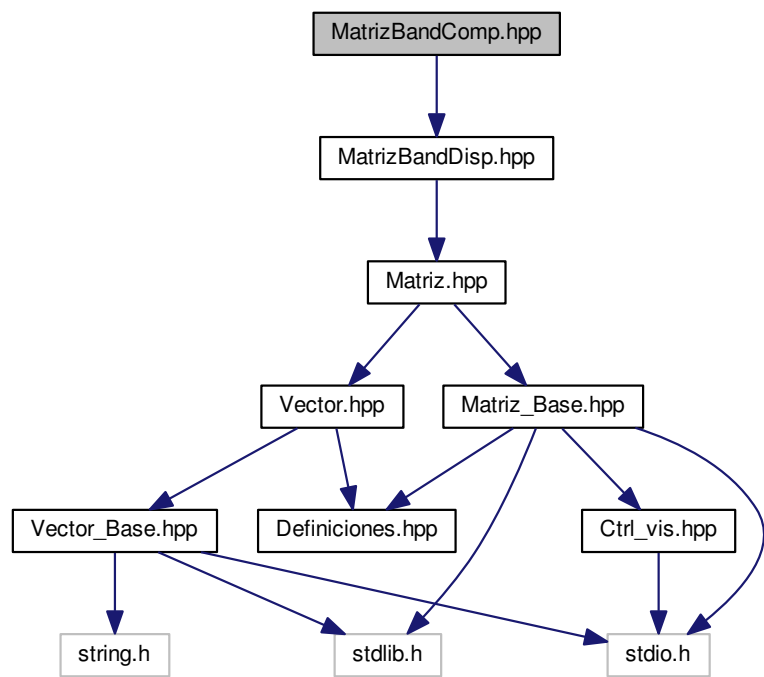
7.12.1 Macro Definition Documentation

- 7.12.1.1 #define EPS 1e-14

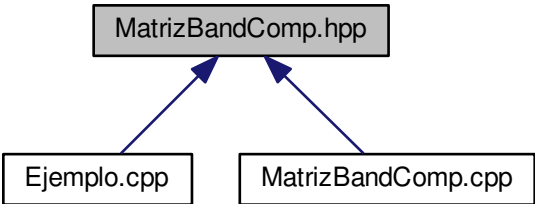
7.13 MatrizBandComp.hpp File Reference

```
#include "MatrizBandDisp.hpp"
```

Include dependency graph for MatrizBandComp.hpp:



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



Classes

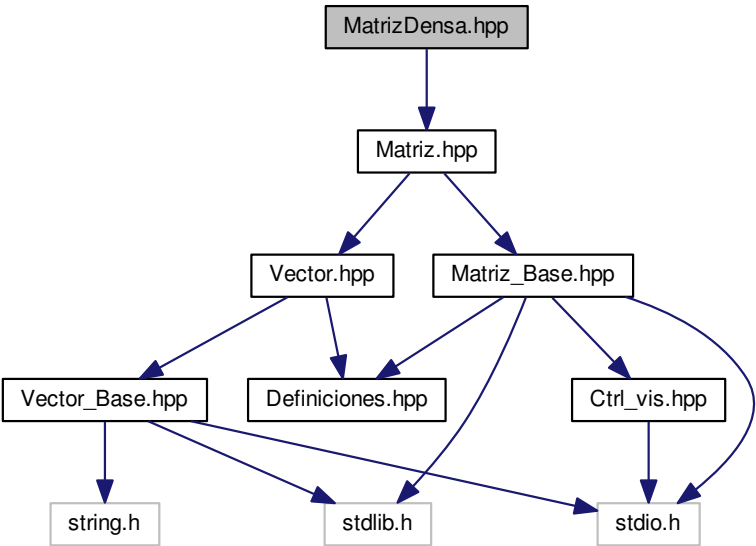
- class [MatrizBandComp](#)

7.14 MatrizBandDisp.hpp File Reference

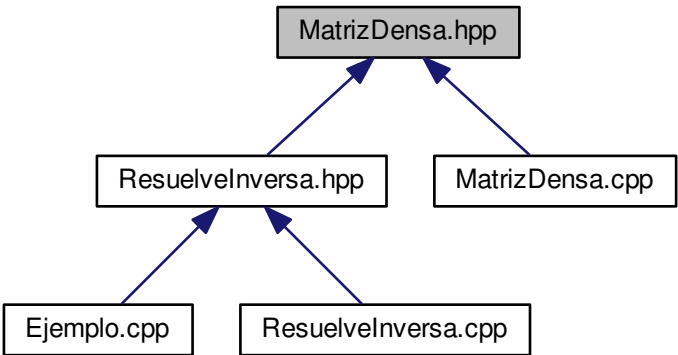
```
#include "Matriz.hpp"
```


7.16 MatrizDensa.hpp File Reference

```
#include "Matriz.hpp"
Include dependency graph for MatrizDensa.hpp:
```



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



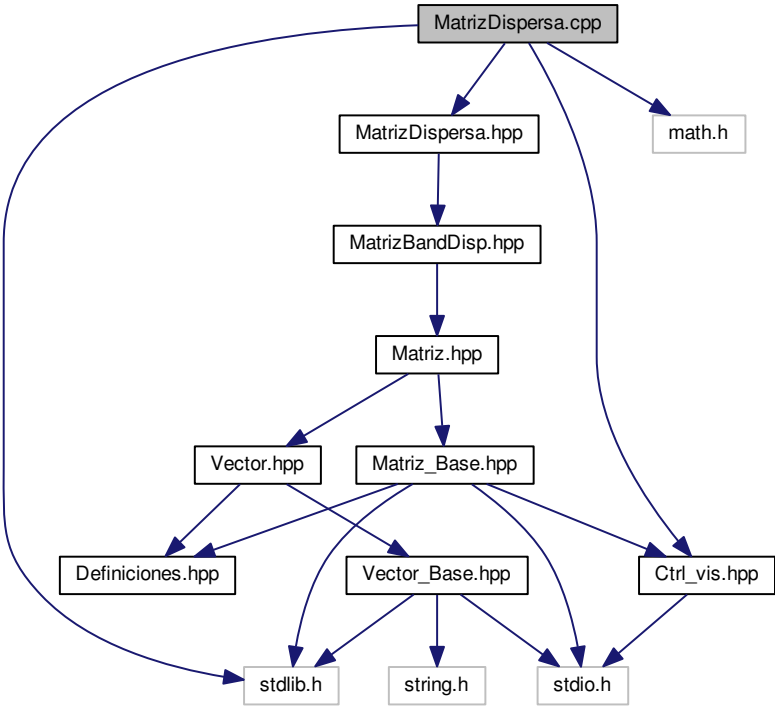
Classes

- class [MatrizDensa](#)
Clase para el trabajar con matrices densas de punto flotante.

7.17 MatrizDispersa.cpp File Reference

```
#include <stdlib.h>
#include <math.h>
#include "MatrizDispersa.hpp"
#include "Ctrl_vis.hpp"
```

Include dependency graph for MatrizDispersa.cpp:



Macros

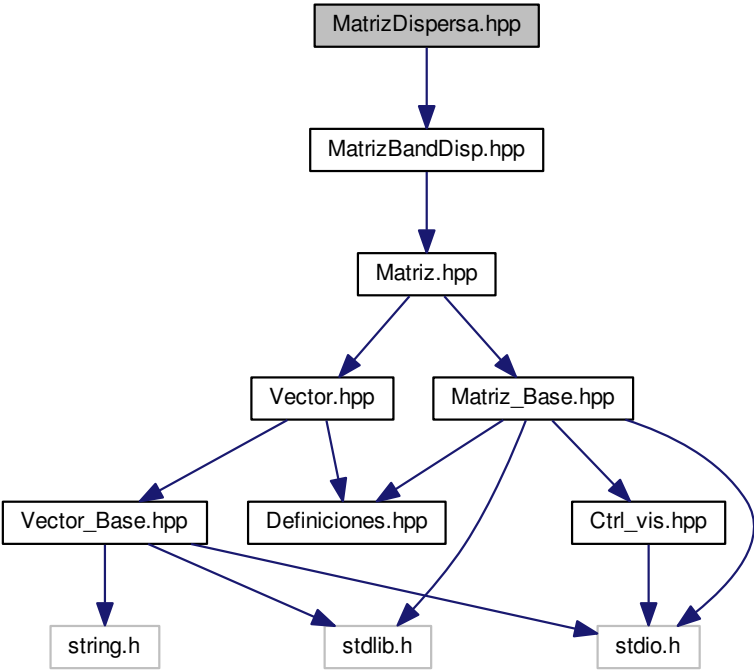
· #define EPS 1e-14

7.17.1 Macro Definition Documentation

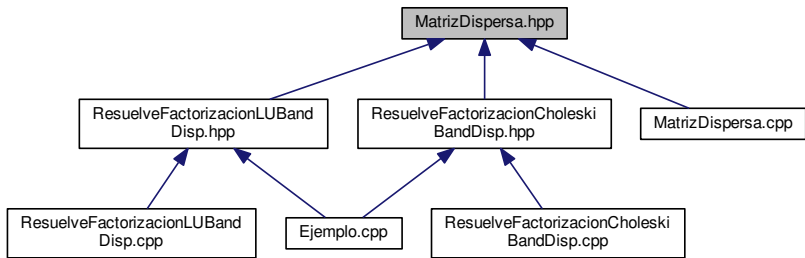
7.17.1.1 #define EPS 1e-14

7.18 MatrizDispersa.hpp File Reference

#include "MatrizBandDisp.hpp"
Include dependency graph for MatrizDispersa.hpp:



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

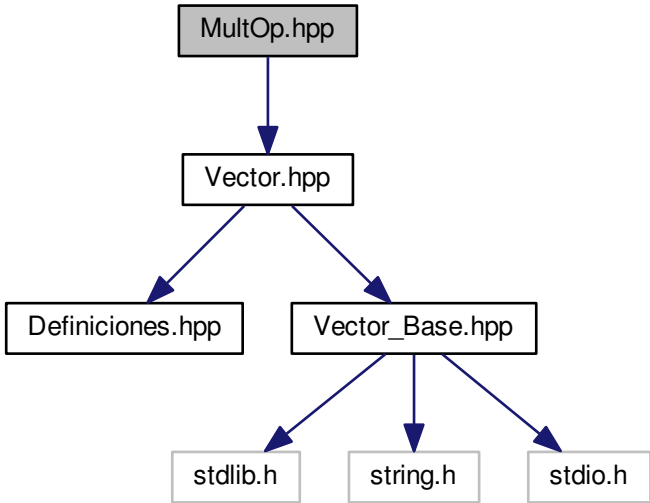


Classes

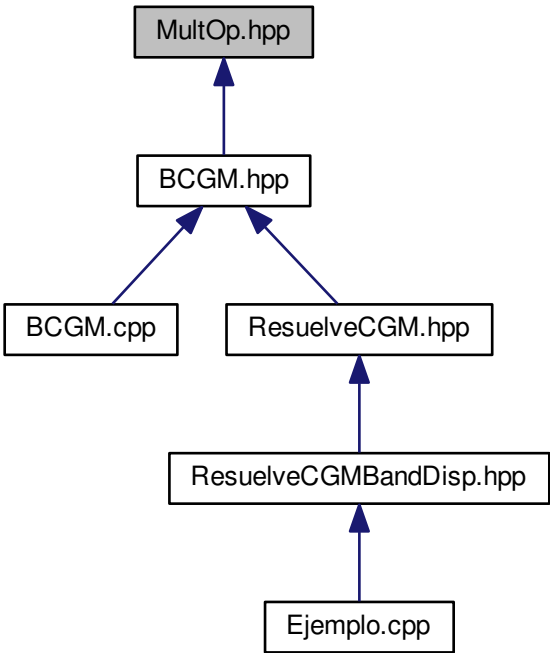
- class [MatrizDispersa](#)

7.19 MultOp.hpp File Reference

`#include "Vector.hpp"`
Include dependency graph for MultOp.hpp:



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

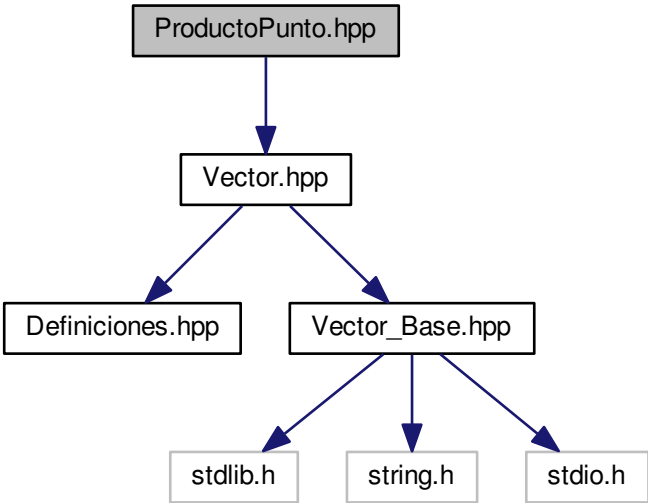


Classes

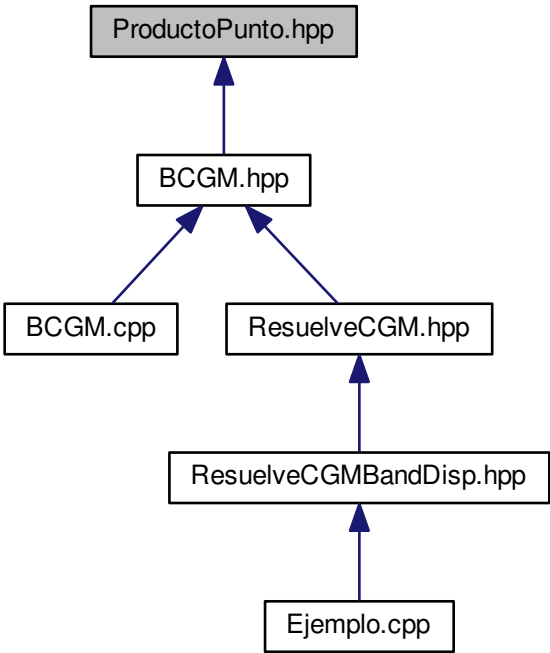
- class [MultOp](#)
Clase para implementar la multiplicaci la matriz por el vector.

7.20 ProductoPunto.hpp File Reference

`#include "Vector.hpp"`
Include dependency graph for **ProductoPunto.hpp**:



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



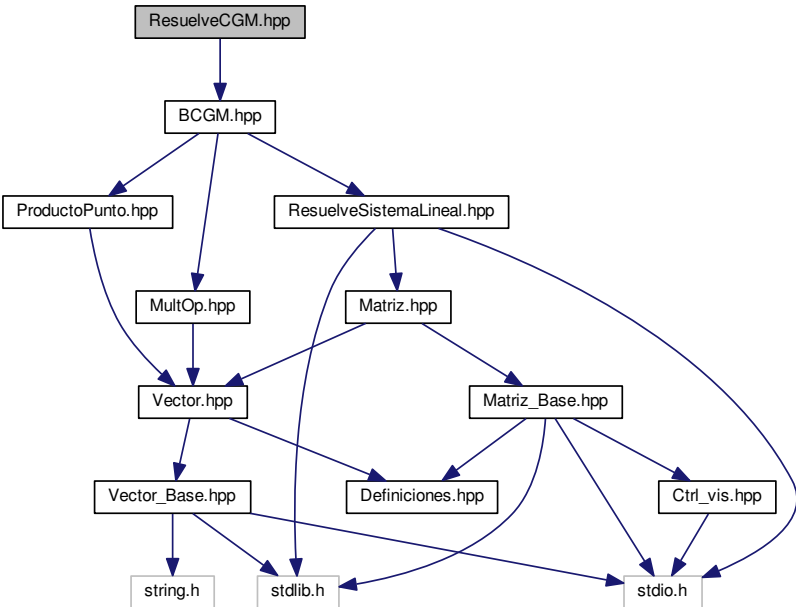
Classes

- class [ProductoPunto](#)
Clase para implementar el producto punto.

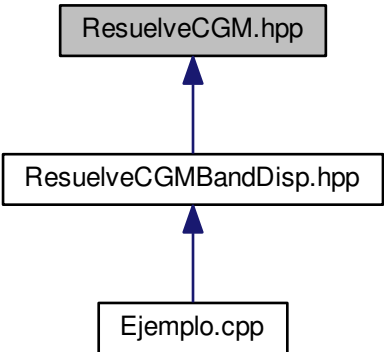
7.21 ResuelveCGM.hpp File Reference

```
#include "BCGM.hpp"
```

Include dependency graph for ResuelveCGM.hpp:



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

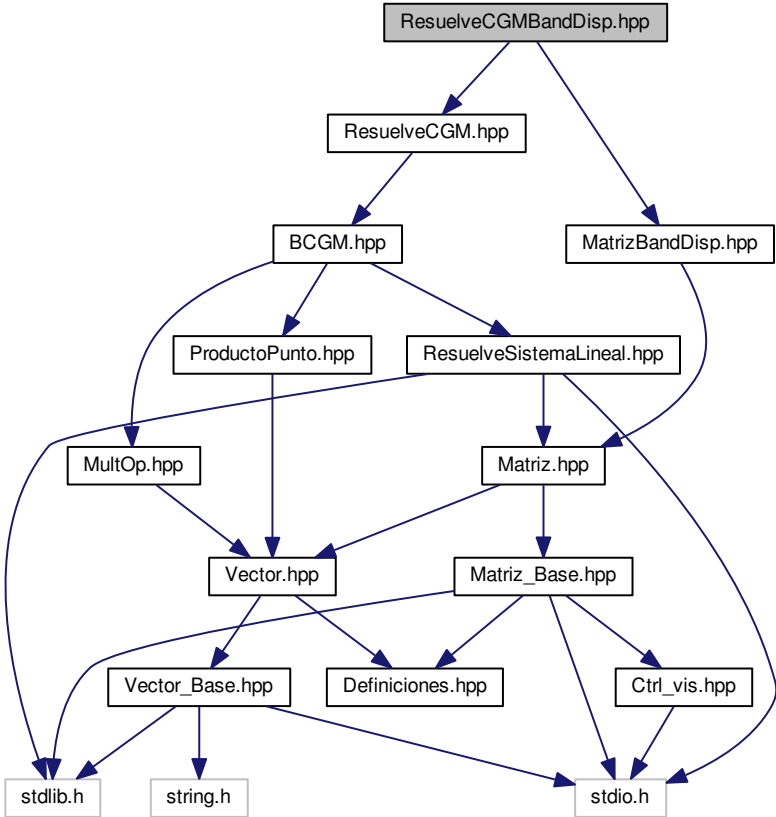


Classes

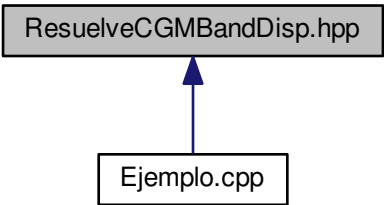
- class [ResuelveCGM](#)
Clase para resolucil sistema lineal mediante CGM standard.

7.22 ResuelveCGMBandDisp.hpp File Reference

```
#include "ResuelveCGM.hpp"
#include "MatrizBandDisp.hpp"
Include dependency graph for ResuelveCGMBandDisp.hpp:
```



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

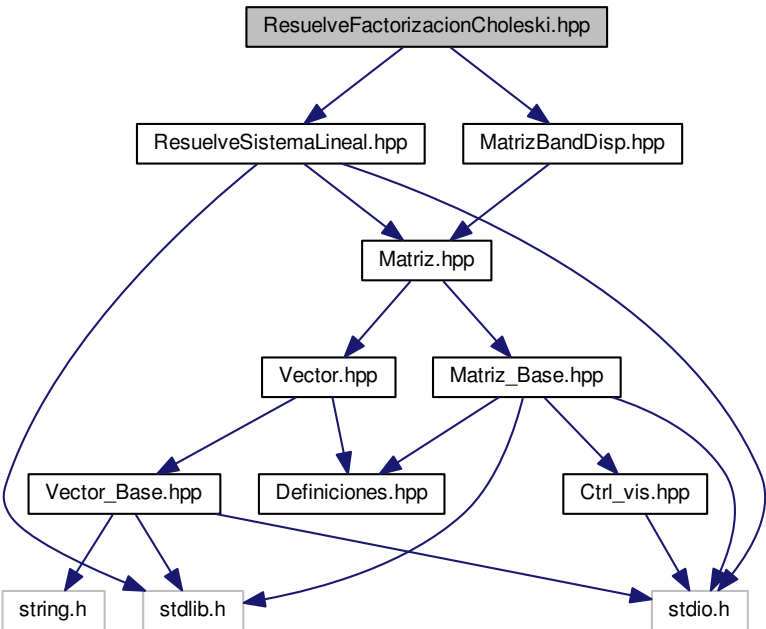


Classes

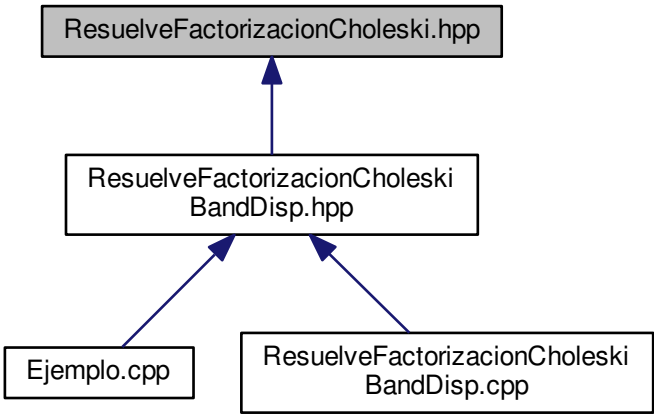
- class [ResuelveCGMBandDisp](#)
Clase para resolucil sistema lineal mediante CGM.

7.23 ResuelveFactorizacionCholeski.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
#include "MatrizBandDisp.hpp"
Include dependency graph for ResuelveFactorizacionCholeski.hpp:
```



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

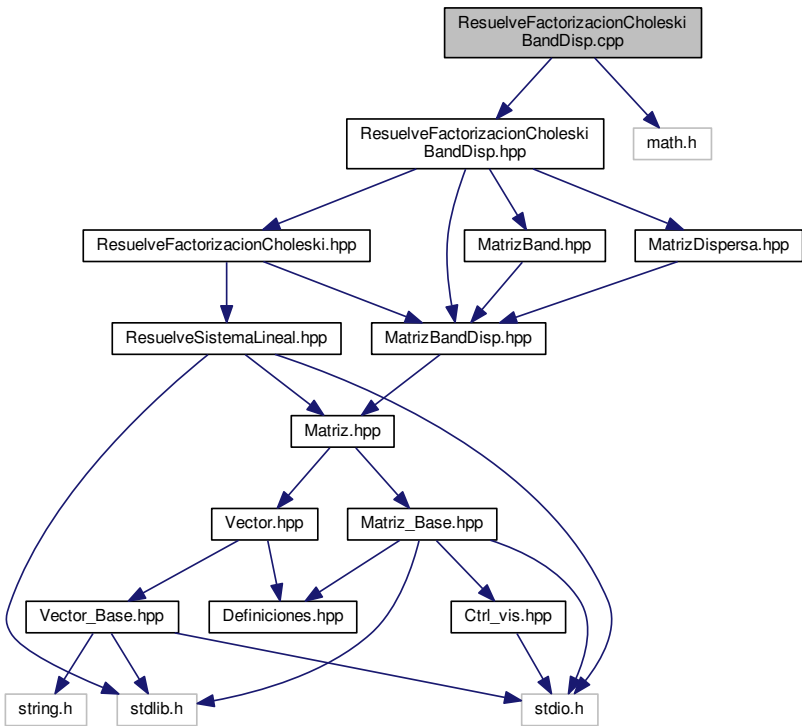


Classes

- class [ResuelveFactorizacionCholeski](#)
Clase para resolucil sistema lineal usando factorizacion Choleski.

7.24 ResuelveFactorizacionCholeskiBandDisp.cpp File Reference

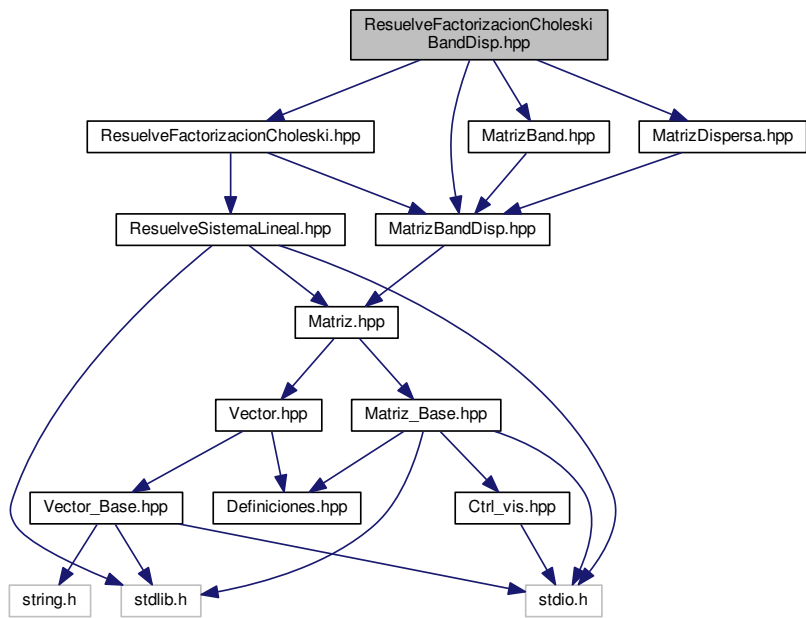
```
#include "ResuelveFactorizacionCholeskiBandDisp.hpp"
#include <math.h>
Include dependency graph for ResuelveFactorizacionCholeskiBandDisp.cpp:
```



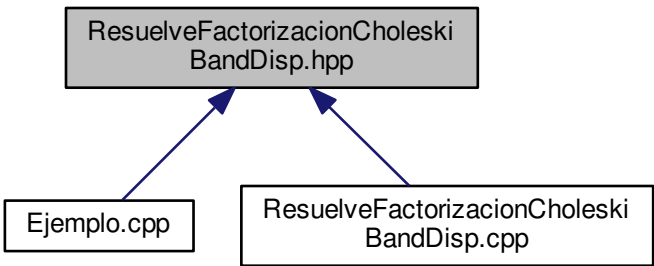
7.25 ResuelveFactorizacionCholeskiBandDisp.hpp File Reference

```
#include "ResuelveFactorizacionCholeski.hpp"
#include "MatrizBandDisp.hpp"
#include "MatrizBand.hpp"
#include "MatrizDispersa.hpp"
```

Include dependency graph for ResuelveFactorizacionCholeskiBandDisp.hpp:



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



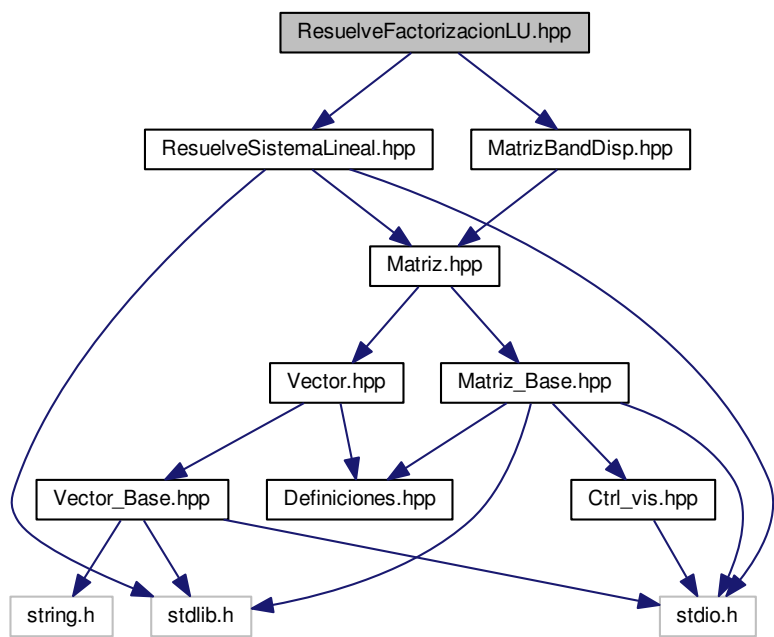
Classes

- class [ResuelveFactorizacionCholeskiBandDisp](#)
Clase para resolucil sistema lineal usando factorizacion Choleski.

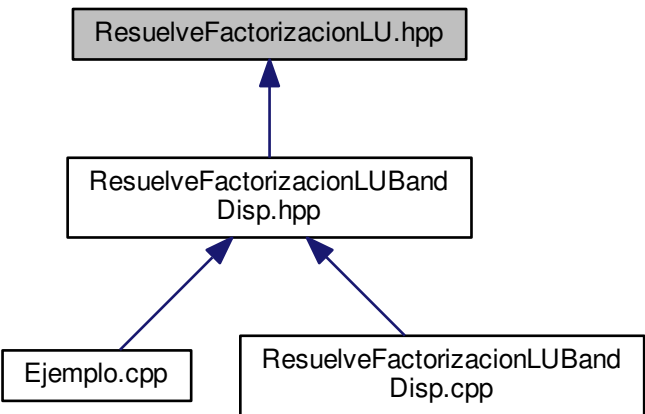
7.26 ResuelveFactorizacionLU.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
#include "MatrizBandDisp.hpp"
```

Include dependency graph for ResuelveFactorizacionLU.hpp:



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



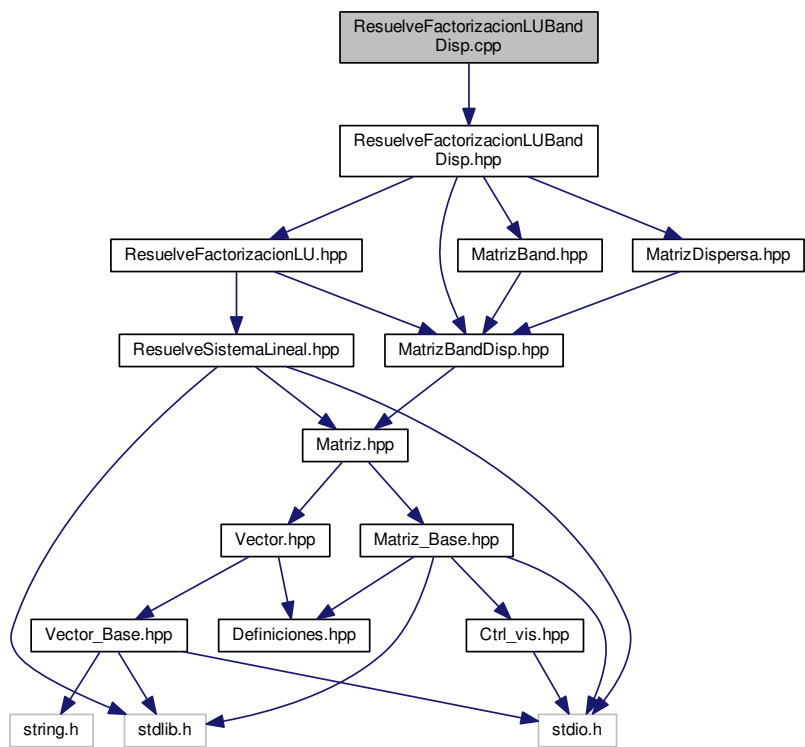
Classes

- class [ResuelveFactorizacionLU](#)
Clase para resoluciil sistema lineal usando factorizacion LU.

7.27 ResuelveFactorizacionLUBandDisp.cpp File Reference

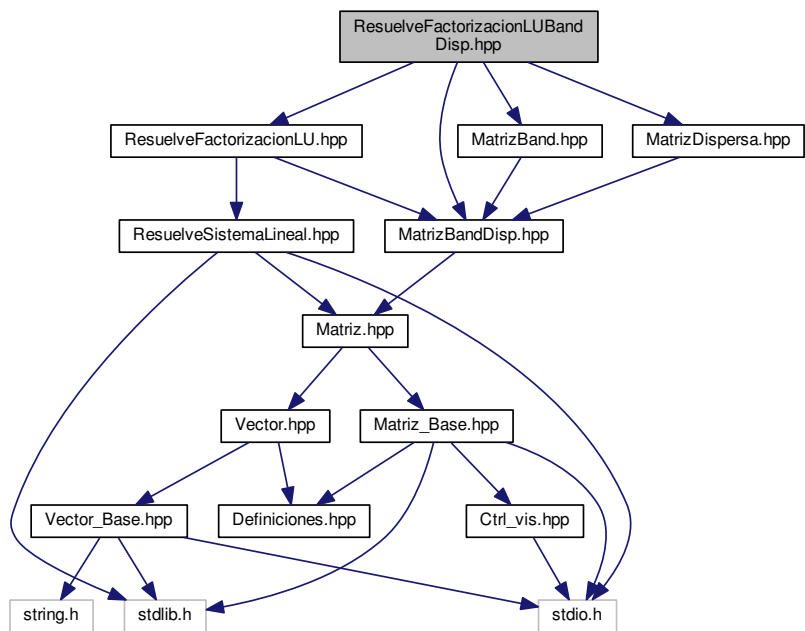
```
#include "ResuelveFactorizacionLUBandDisp.hpp"
```


Include dependency graph for ResuelveFactorizacionLUBandDisp.cpp:

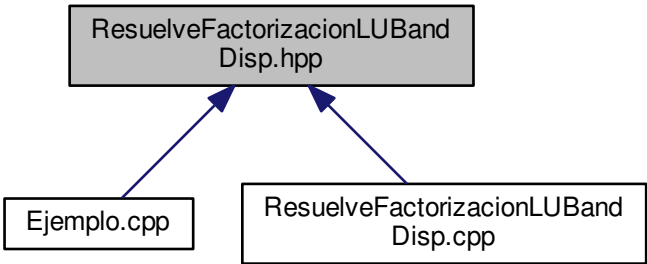


7.28 ResuelveFactorizacionLUBandDisp.hpp File Reference

```
#include "ResuelveFactorizacionLU.hpp"
#include "MatrizBandDisp.hpp"
#include "MatrizBand.hpp"
#include "MatrizDispersa.hpp"
Include dependency graph for ResuelveFactorizacionLUBandDisp.hpp:
```



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

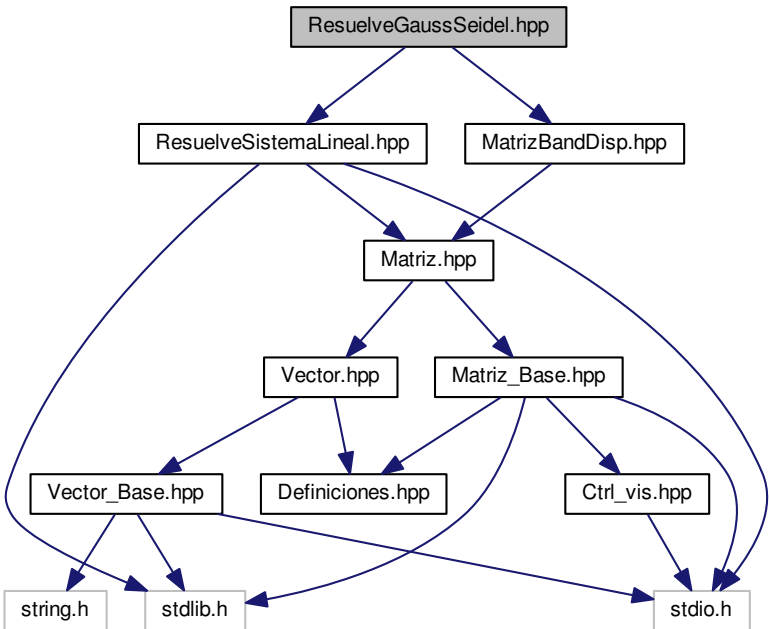


Classes

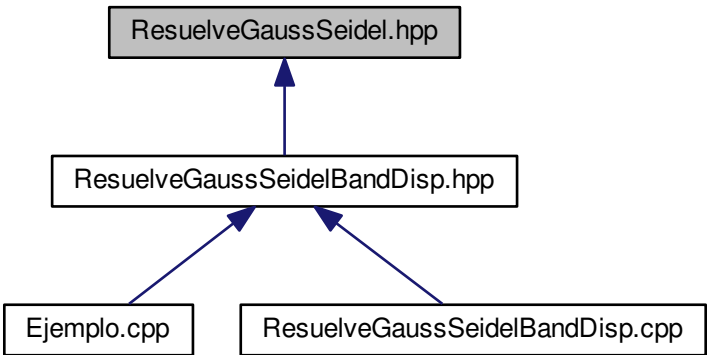
- class [ResuelveFactorizacionLUBandDisp](#)
Clase para resoluci sistema lineal usando factorizacion LU.

7.29 ResuelveGaussSeidel.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
#include "MatrizBandDisp.hpp"
Include dependency graph for ResuelveGaussSeidel.hpp:
```



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

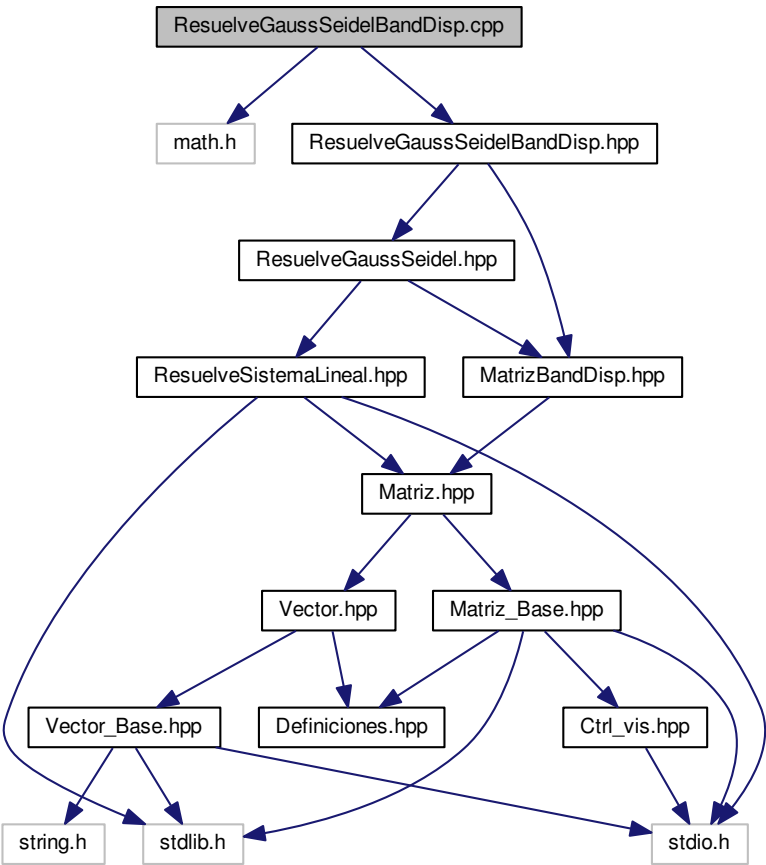


Classes

- class [ResuelveGaussSeidel](#)
Clase para resoluci3n sistema lineal mediante Gauss-Seidel.

7.30 ResuelveGaussSeidelBandDisp.cpp File Reference

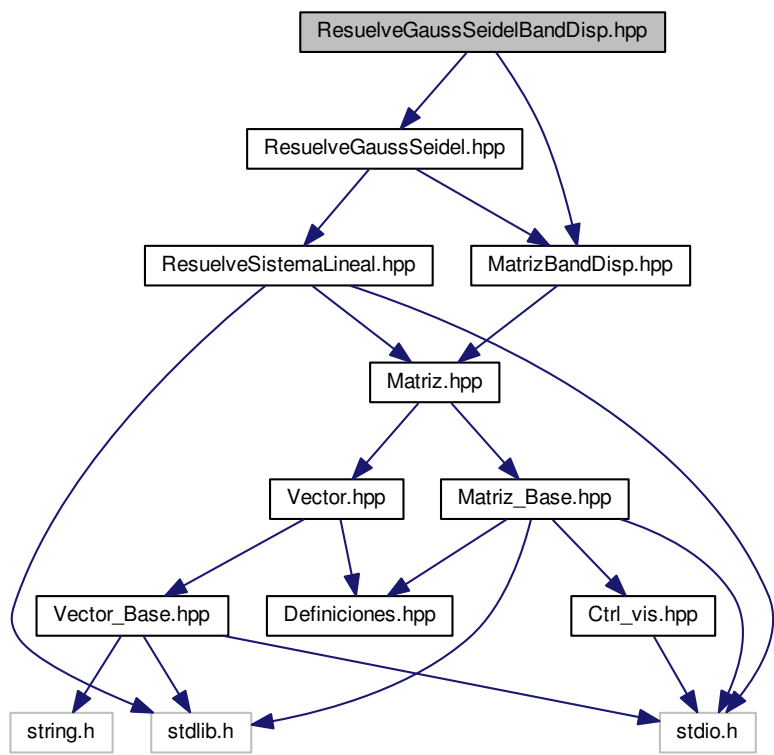
```
#include <math.h>
#include "ResuelveGaussSeidelBandDisp.hpp"
Include dependency graph for ResuelveGaussSeidelBandDisp.cpp:
```



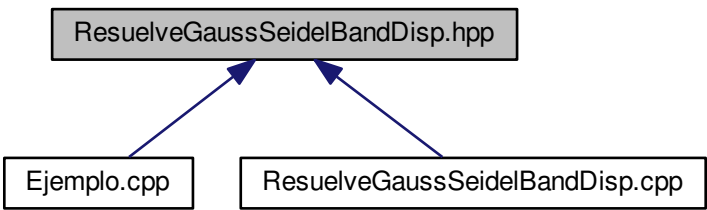
7.31 ResuelveGaussSeidelBandDisp.hpp File Reference

```
#include "ResuelveGaussSeidel.hpp"
#include "MatrizBandDisp.hpp"
```

Include dependency graph for ResuelveGaussSeidelBandDisp.hpp:



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



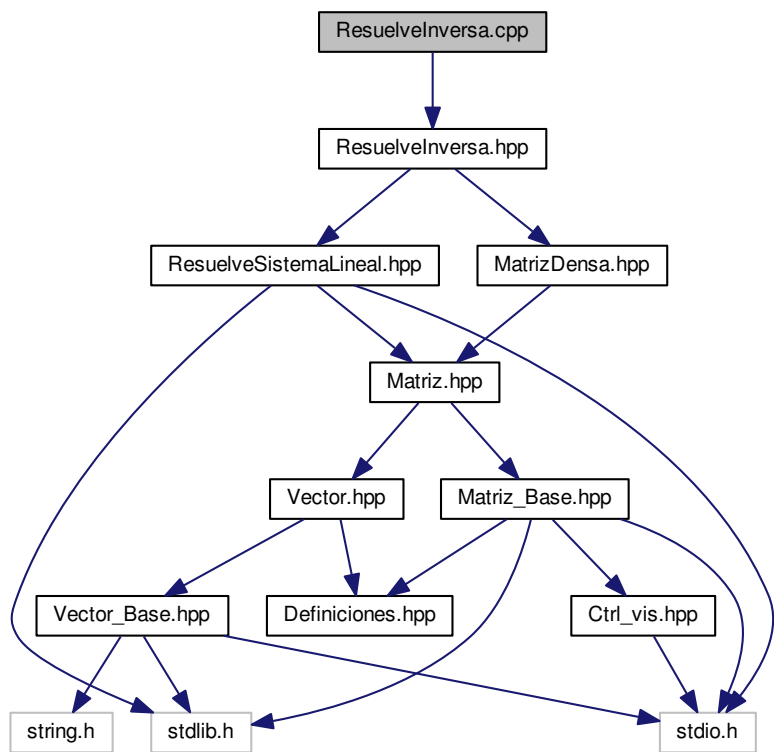
Classes

- class [ResuelveGaussSeidelBandDisp](#)
Clase para resolucil sistema lineal mediante Gauss-Seidel.

7.32 ResuelveInversa.cpp File Reference

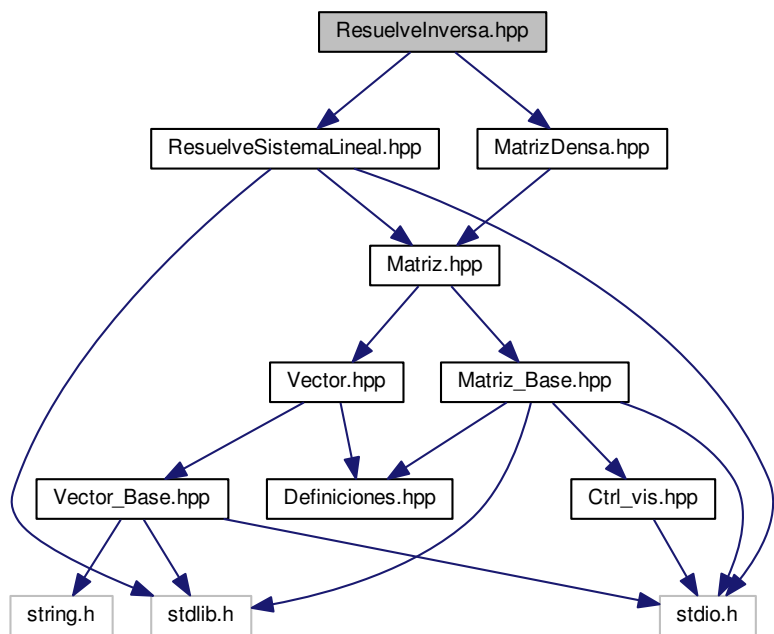
```
#include "ResuelveInversa.hpp"
```

Include dependency graph for ResuelveInversa.cpp:

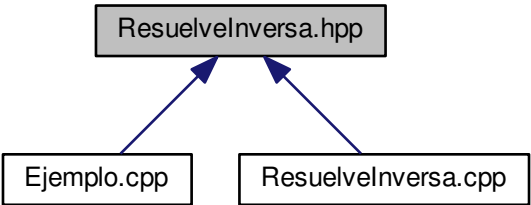


7.33 ResuelveInversa.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
#include "MatrizDensa.hpp"
Include dependency graph for ResuelveInversa.hpp:
```



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

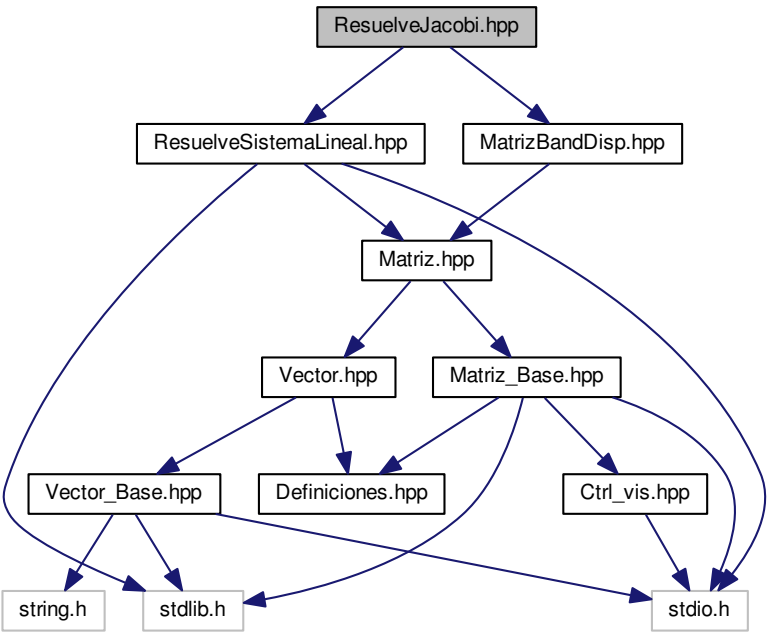


Classes

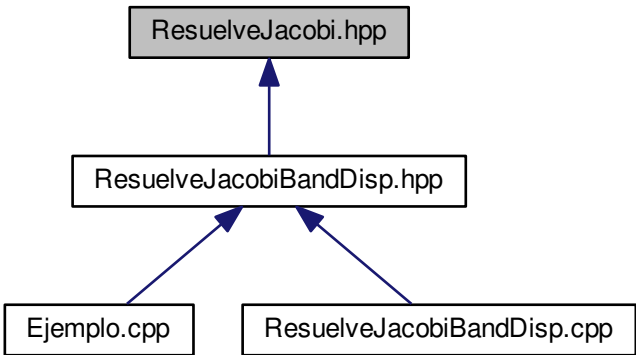
- class [ResuelveInversa](#)
Clase para resoluci sistema lineal mediante el uso de la matriz inversa.

7.34 ResuelveJacobi.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
#include "MatrizBandDisp.hpp"
Include dependency graph for ResuelveJacobi.hpp:
```



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

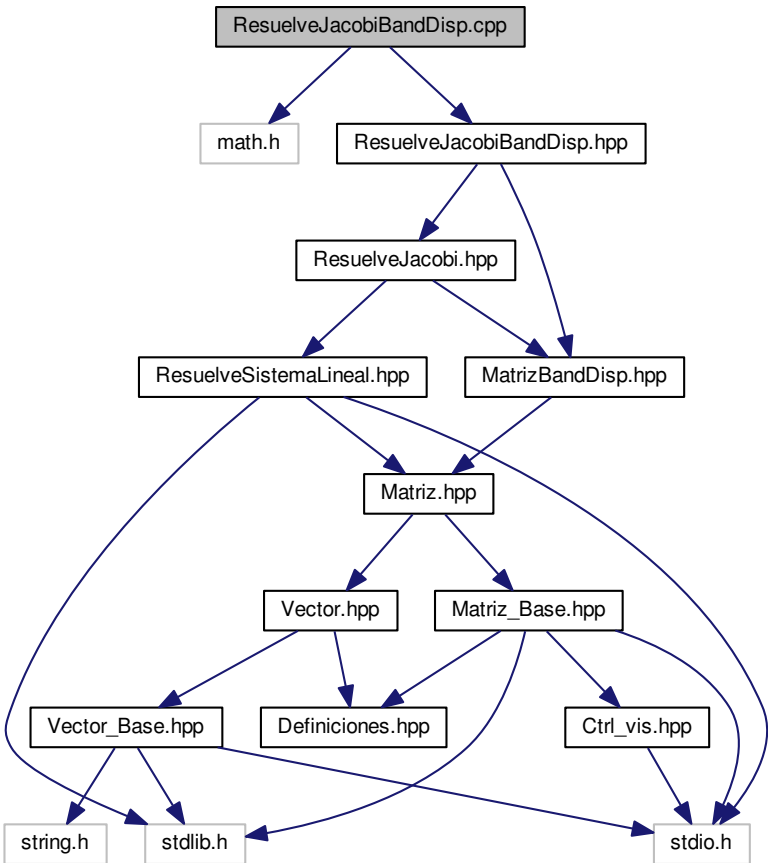


Classes

- class [ResuelveJacobi](#)
Clase para resoluci3n de sistema lineal mediante Jacobi.

7.35 ResuelveJacobiBandDisp.cpp File Reference

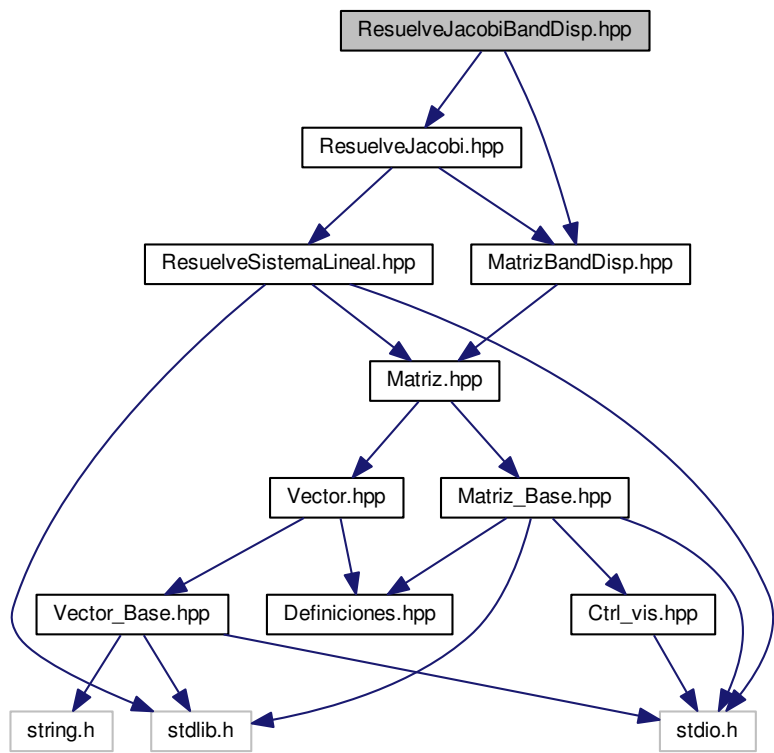
```
#include <math.h>
#include "ResuelveJacobiBandDisp.hpp"
Include dependency graph for ResuelveJacobiBandDisp.cpp:
```



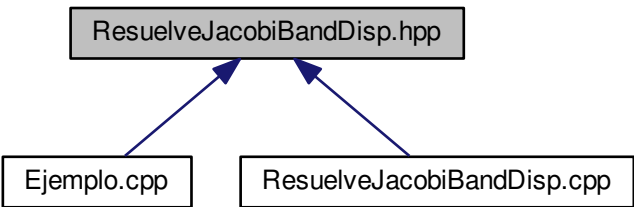
7.36 ResuelveJacobiBandDisp.hpp File Reference

```
#include "ResuelveJacobi.hpp"
#include "MatrizBandDisp.hpp"
```

Include dependency graph for ResuelveJacobiBandDisp.hpp:



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



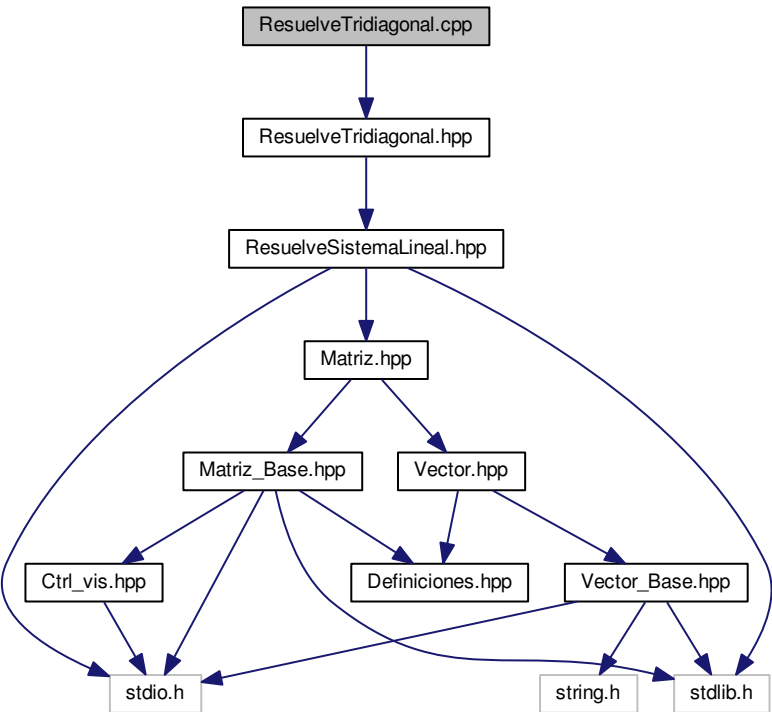
Classes

- class [ResuelveJacobiBandDisp](#)
Clase para resolucil sistema lineal mediante Jacobi.

7.37 ResuelveSistemaLineal.hpp File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include "Matriz.hpp"
```

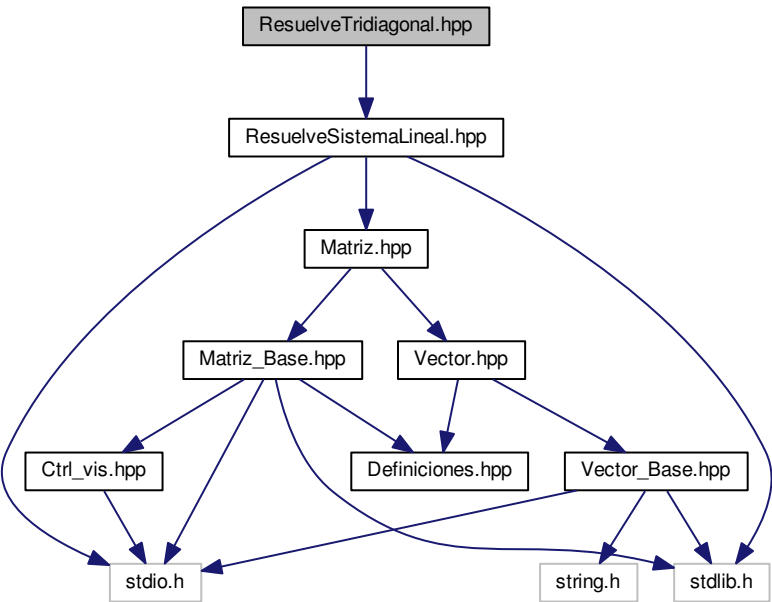

Include dependency graph for ResuelveTridiagonal.cpp:



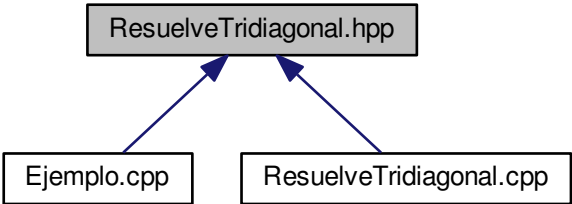
7.39 ResuelveTridiagonal.hpp File Reference

```
#include "ResuelveSistemaLineal.hpp"
```

Include dependency graph for ResuelveTridiagonal.hpp:



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

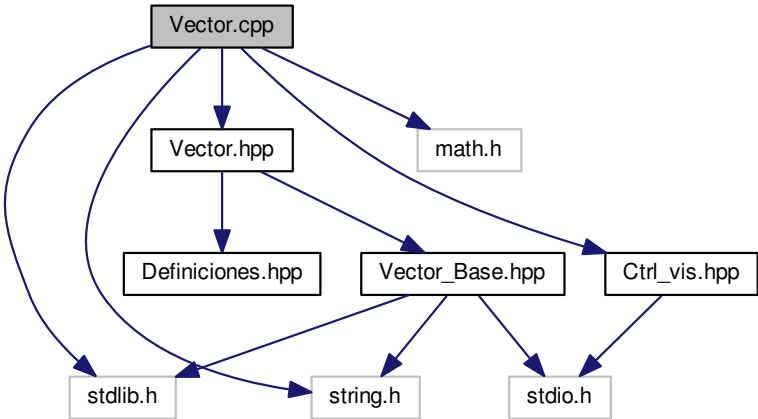


Classes

- class [ResuelveTridiagonal](#)
Clase para resoluci3n sistema lineal tridiagonal.

7.40 Vector.cpp File Reference

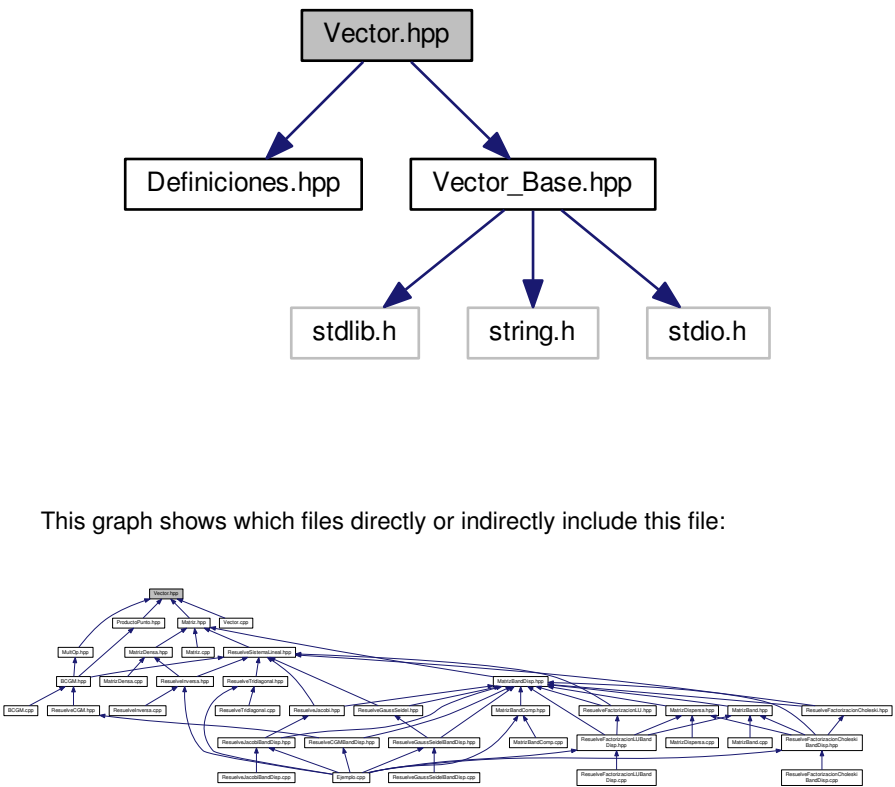
```
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "Vector.hpp"
#include "Ctrl_vis.hpp"
Include dependency graph for Vector.cpp:
```



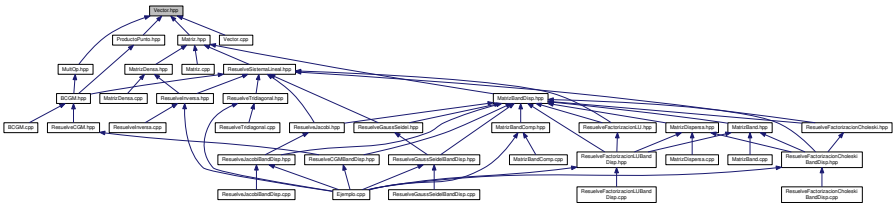
7.41 Vector.hpp File Reference

```
#include "Definiciones.hpp"
#include "Vector_Base.hpp"
```

Include dependency graph for Vector.hpp:



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

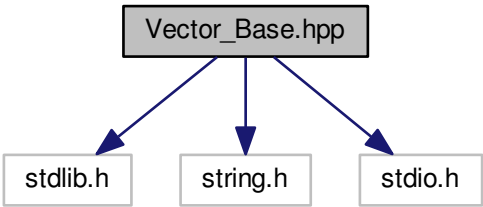


Classes

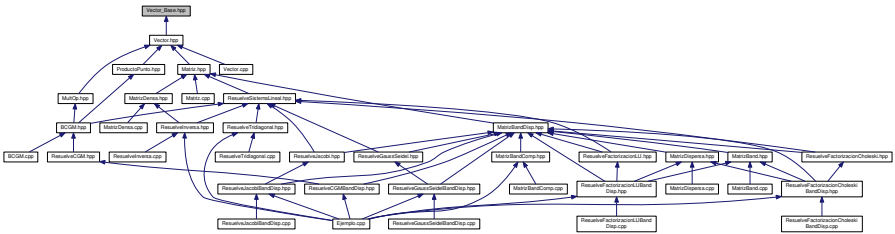
- class [Vector](#)
Clase para el trabajar con vectores de punto flotante.

7.42 Vector_Base.hpp File Reference

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
Include dependency graph for Vector_Base.hpp:
```



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



Classes

- class [Vector_Base](#)

Clase base para el trabajar con vectores.

Chapter 8

Example Documentation

8.1 EjemploMatrizBand.cpp

Esta clase implementa los componentes para el trabajar con matrices bandadas de punto flotante

8.2 EjemploMatrizBandComp.cpp

Esta clase implementa los componentes para el trabajar con matrices bandadas de punto flotante que puede ser compactada

8.3 EjemploMatrizDensa.cpp

Esta clase implementa los componentes para el trabajar con matrices densas de punto flotante

8.4 EjemploMatrizDispersa.cpp

Esta clase implementa los componentes para el trabajar con matrices dispersas de punto flotante

8.5 EjemploResuelveSistemaLineal.cpp

Esta clase implementa los componentes para la resolucil sistema lineal mediante CGM

8.6 EjemploVector.cpp

Esta clase implementa los comportamientos momunes para el trabajar con vectores de punto flotante