

[Help](#)

```
#ifndef GRIDSPARSE_CONSTRUCTOR
#define GRIDSPARSE_CONSTRUCTOR

#include "pnl/pnl_vector.h"
#include "pnl/pnl_matrix.h"
#include "
href../../../../common/math/equity_pricer/pde_tools_h_src.pdfpde_tools.h"

typedef struct GridSparse
{
    int dim; /*!< dimension of the grid */
    int lev; /*!< level of the grid */
    int size; /*!< size of the grid */
    PnlVectInt *size_in_level; /*!< size of the grid of level d */
    PnlHmatInt *Ind_Father; /*!< Give Index of father [Dimension][Points][LeftOrR
    PnlHmatInt *Ind_Son; /*!< Give Index of Son [Dimension][Points][LeftOrR
    PnlHmatInt *Ind_Neigh; /*!< Give Index of Neighbour [Dimension][Points][Left
    /* PnlMatInt * Ind_Next; /*!< Give Index of Next [Dimension][Points] */
    PnlMatInt *Points; /*!< Give Vector at [Points] as col of Points Points[i
    /*!< Give index on diadic grid of i eme grid points in direction dim. */
    /*PnlHmat * Point_Step; /*!< Give Step for finite difference operator in[Dime
    PremiaPDEDimBoundary *Bnd;
} GridSparse;

extern GridSparse *grid_sparse_create01(int dim, int lev);
extern GridSparse *grid_sparse_create(const PnlVect *X0, const PnlVect *X1, int
extern void GridSparse_free(GridSparse **G);
extern void GridSparse_check_relation(GridSparse *G);
#endif
```