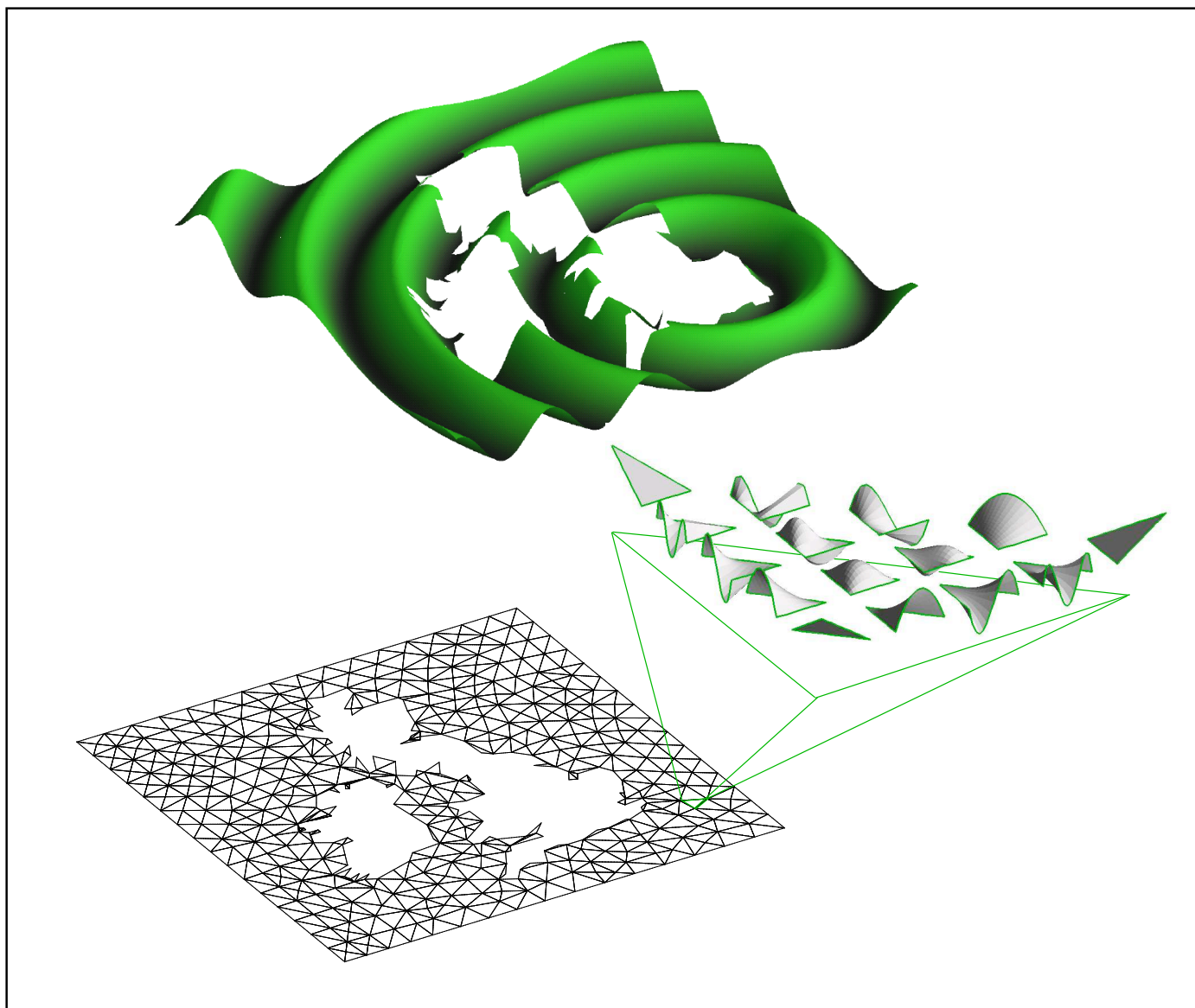


Computing Lab 3



Nektar++

- <http://www.nektar.info>
- a 2D Helmholtz solver in Nektar++
 - input
 - solver
 - output

Nektar++: input

- problem

$$\nabla^2 u - \lambda u = -f$$

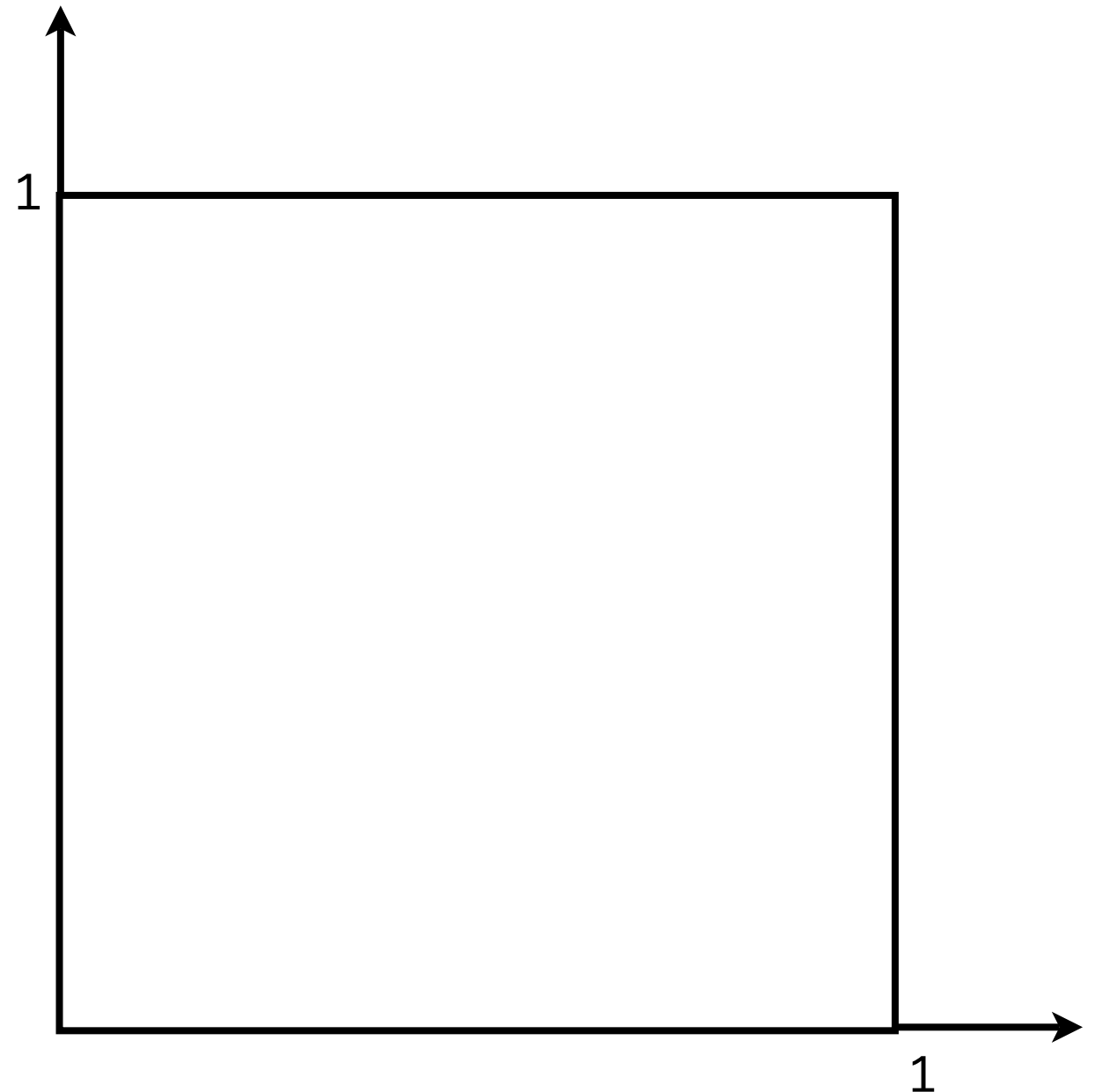
where

$$f = (\lambda + 2\pi^2) \cos(\pi x) \cos(\pi y)$$

$$\lambda = 1$$

boundary conditions

$$g(x, y) = \cos(\pi x) \cos(\pi y) \quad \text{on } \Gamma$$



Nektar++: input

- problem

$$\nabla^2 u - \lambda u = -f$$

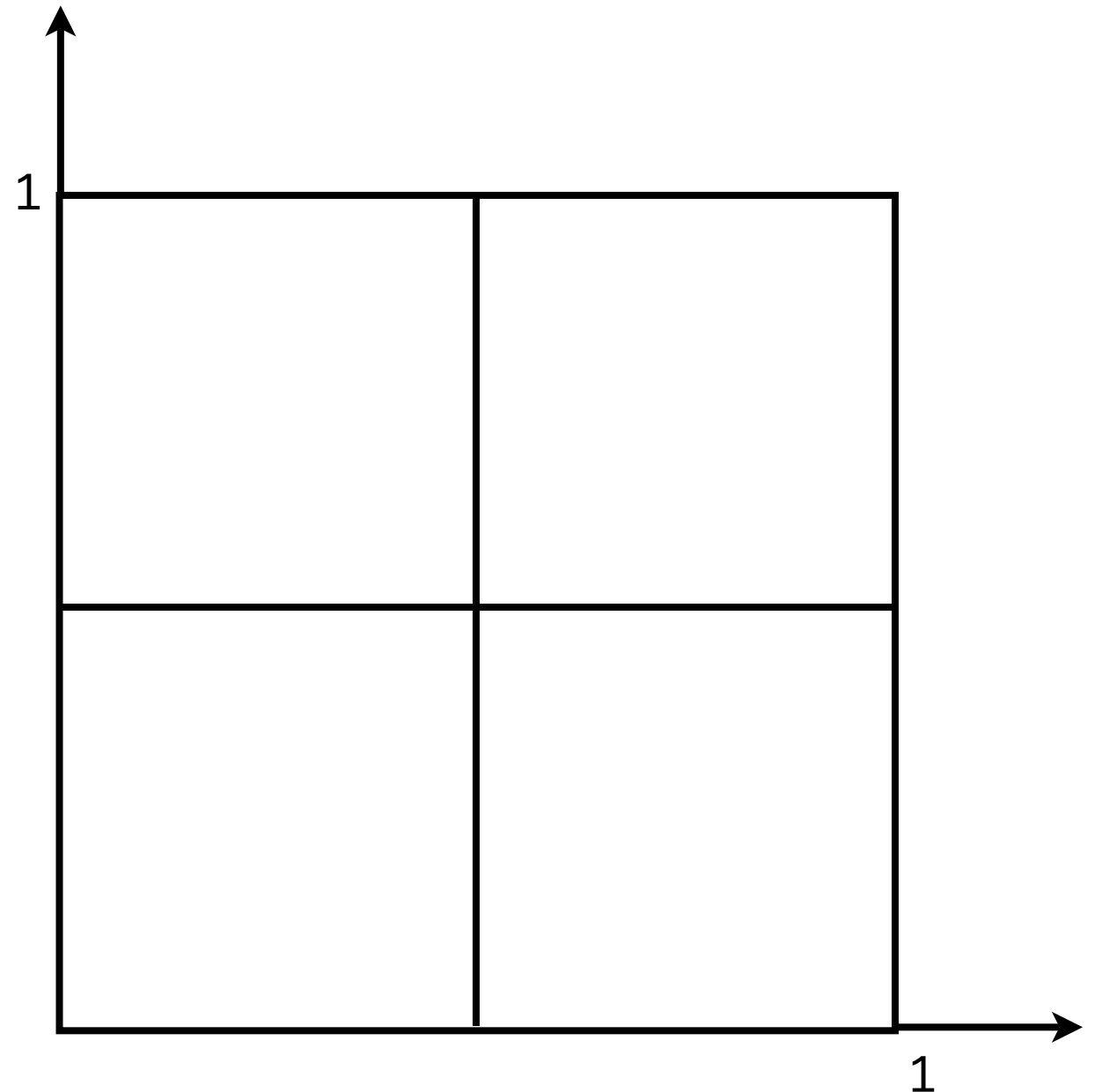
where

$$f = (\lambda + 2\pi^2) \cos(\pi x) \cos(\pi y)$$

$$\lambda = 1$$

boundary conditions

$$g(x, y) = \cos(\pi x) \cos(\pi y) \quad \text{on } \Gamma$$



Nektar++: input

- problem

$$\nabla^2 u - \lambda u = -f$$

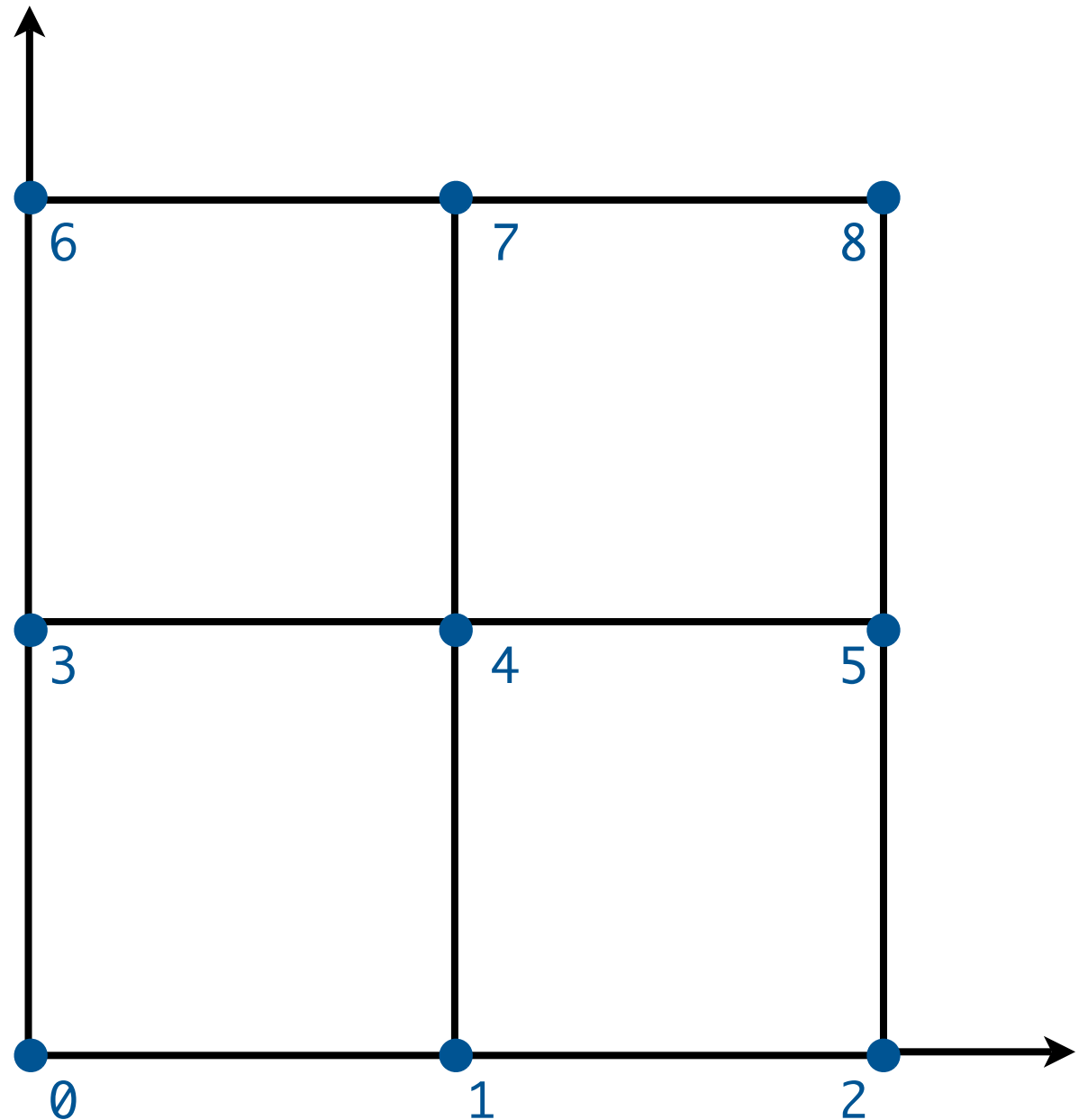
where

$$f = (\lambda + 2\pi^2) \cos(\pi x) \cos(\pi y)$$

$$\lambda = 1$$

boundary conditions

$$g(x, y) = \cos(\pi x) \cos(\pi y) \quad \text{on } \Gamma$$



Nektar++: input

- problem

$$\nabla^2 u - \lambda u = -f$$

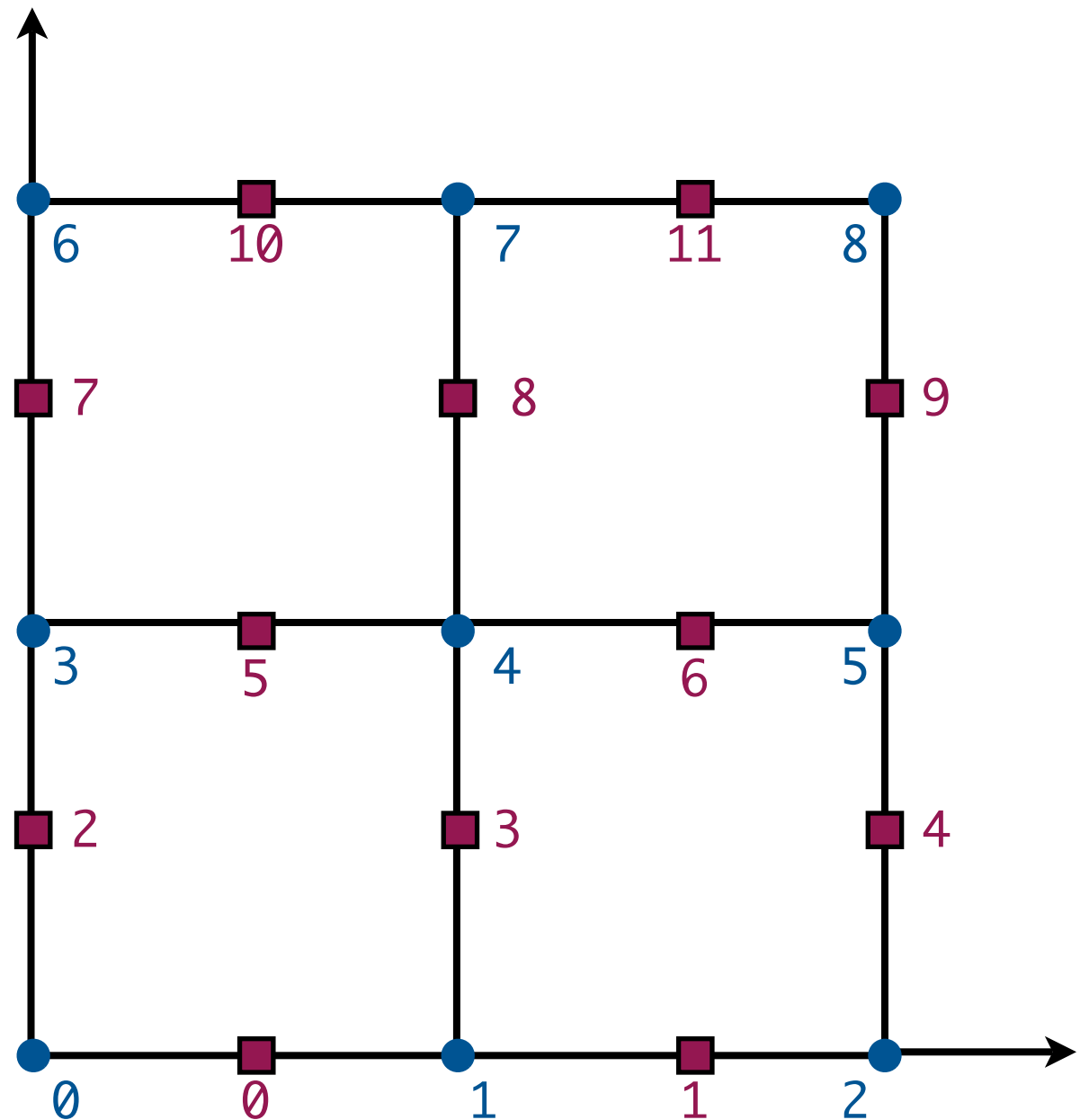
where

$$f = (\lambda + 2\pi^2) \cos(\pi x) \cos(\pi y)$$

$$\lambda = 1$$

boundary conditions

$$g(x, y) = \cos(\pi x) \cos(\pi y) \quad \text{on } \Gamma$$



Nektar++: input

- problem

$$\nabla^2 u - \lambda u = -f$$

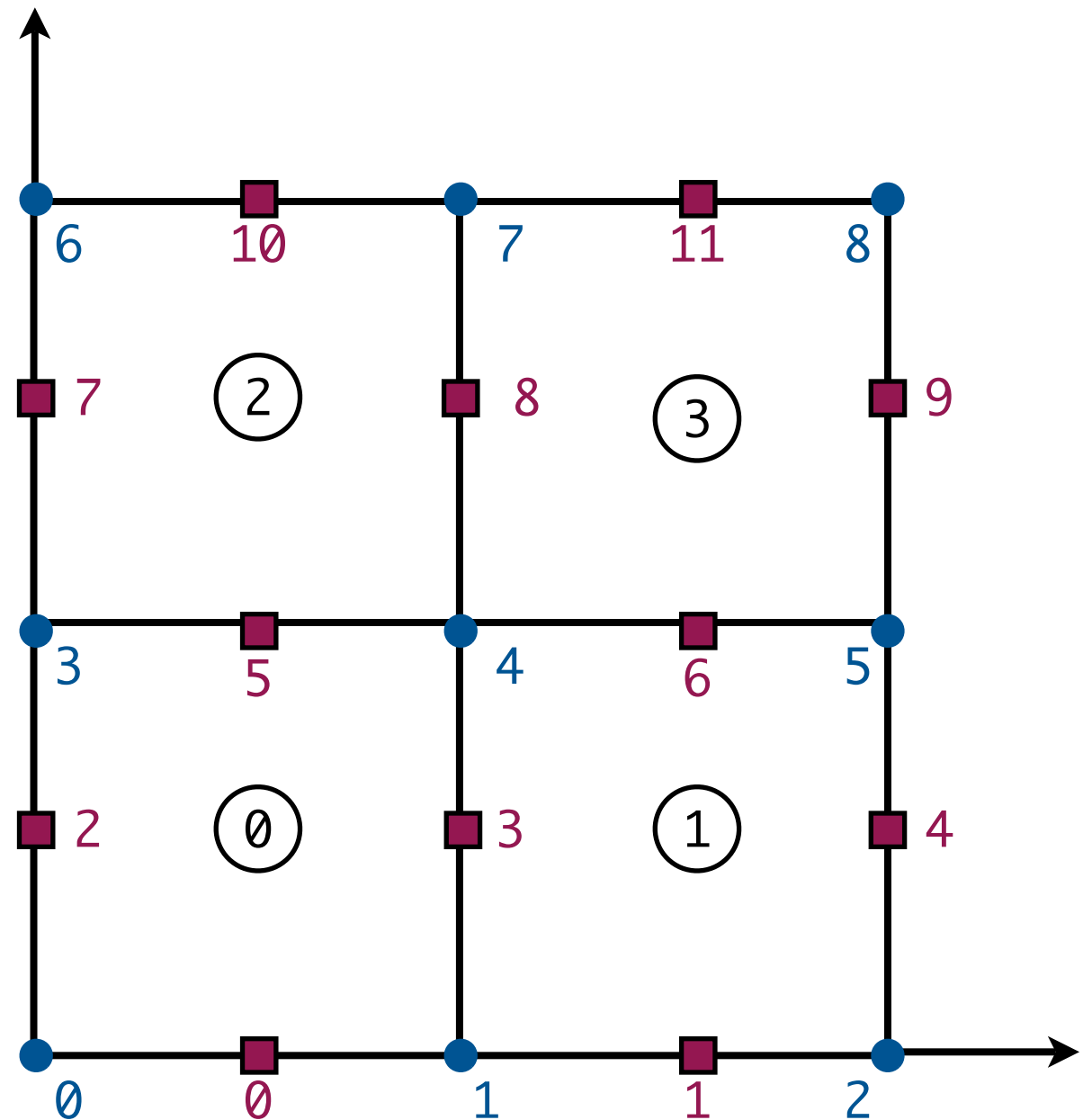
where

$$f = (\lambda + 2\pi^2) \cos(\pi x) \cos(\pi y)$$

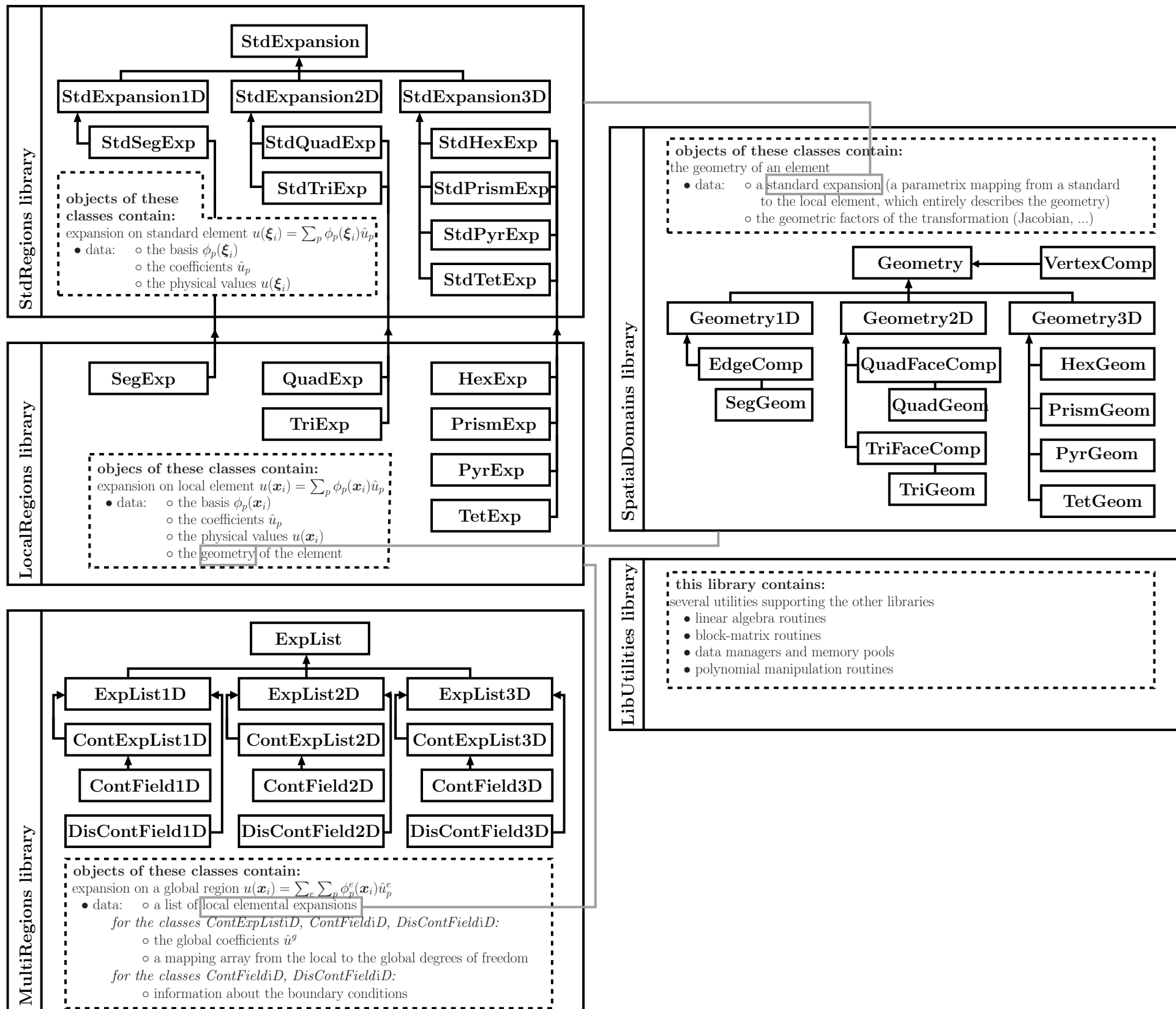
$$\lambda = 1$$

boundary conditions

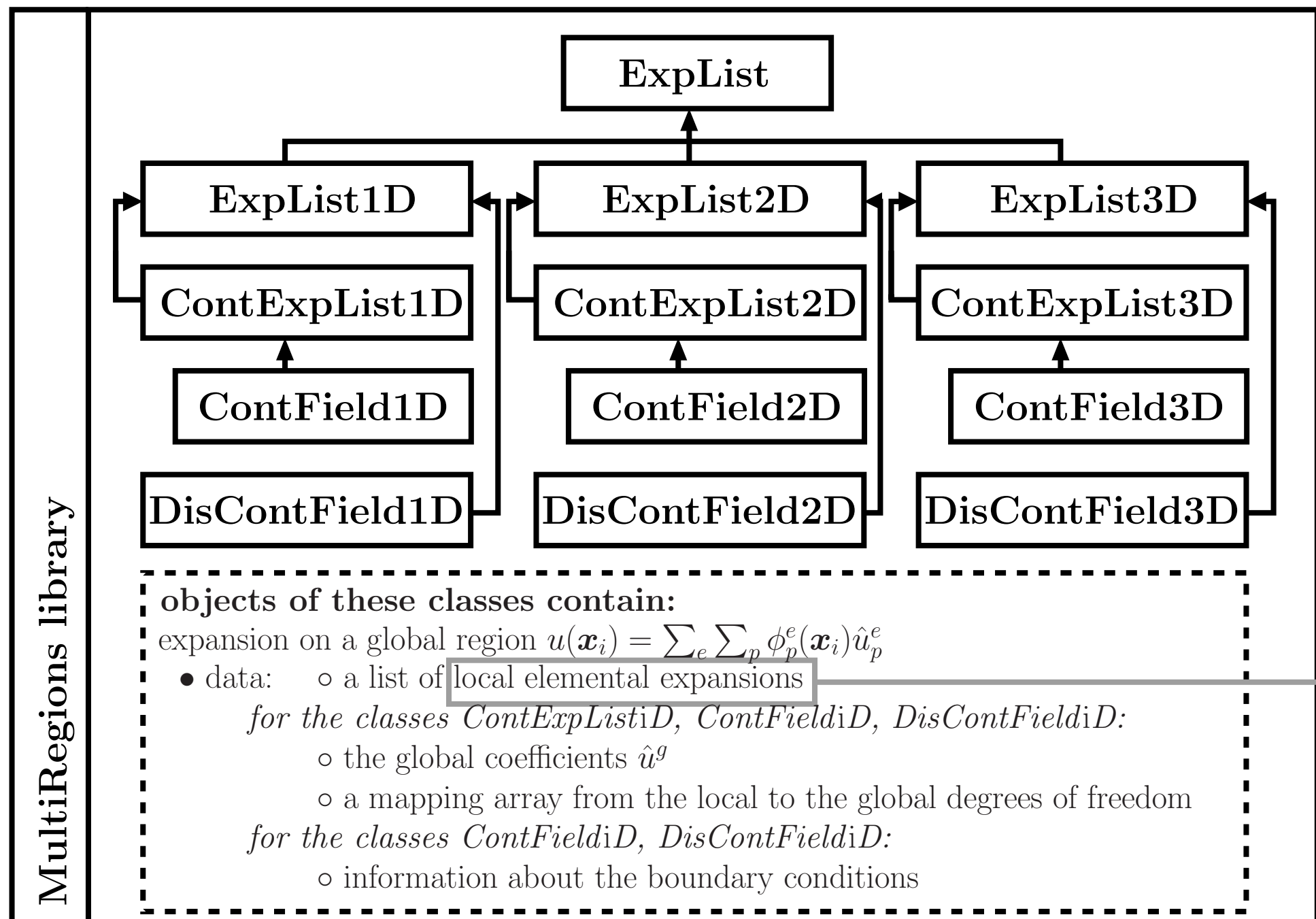
$$g(x, y) = \cos(\pi x) \cos(\pi y) \quad \text{on } \Gamma$$



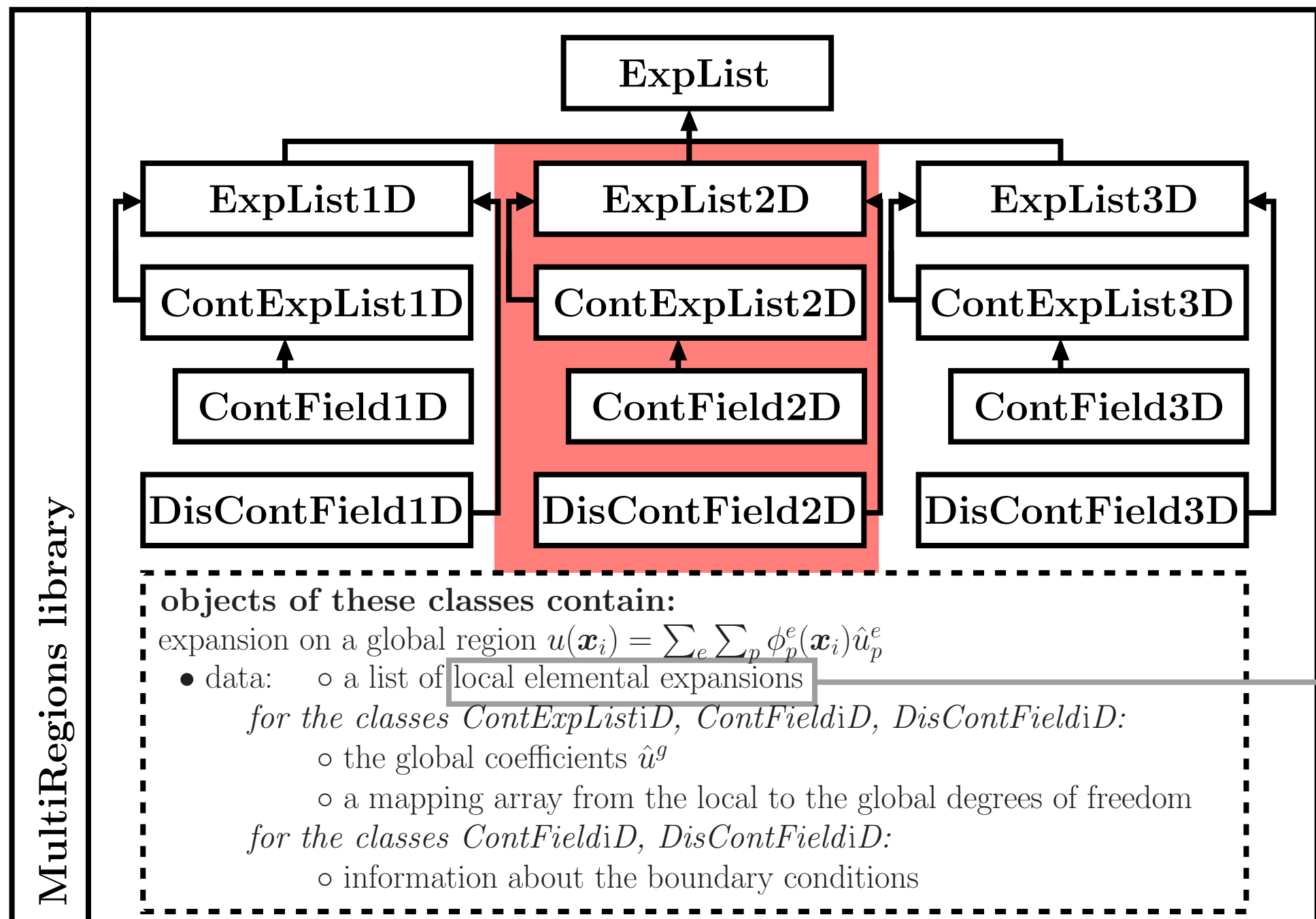
Nektar++: Helmholtz solver



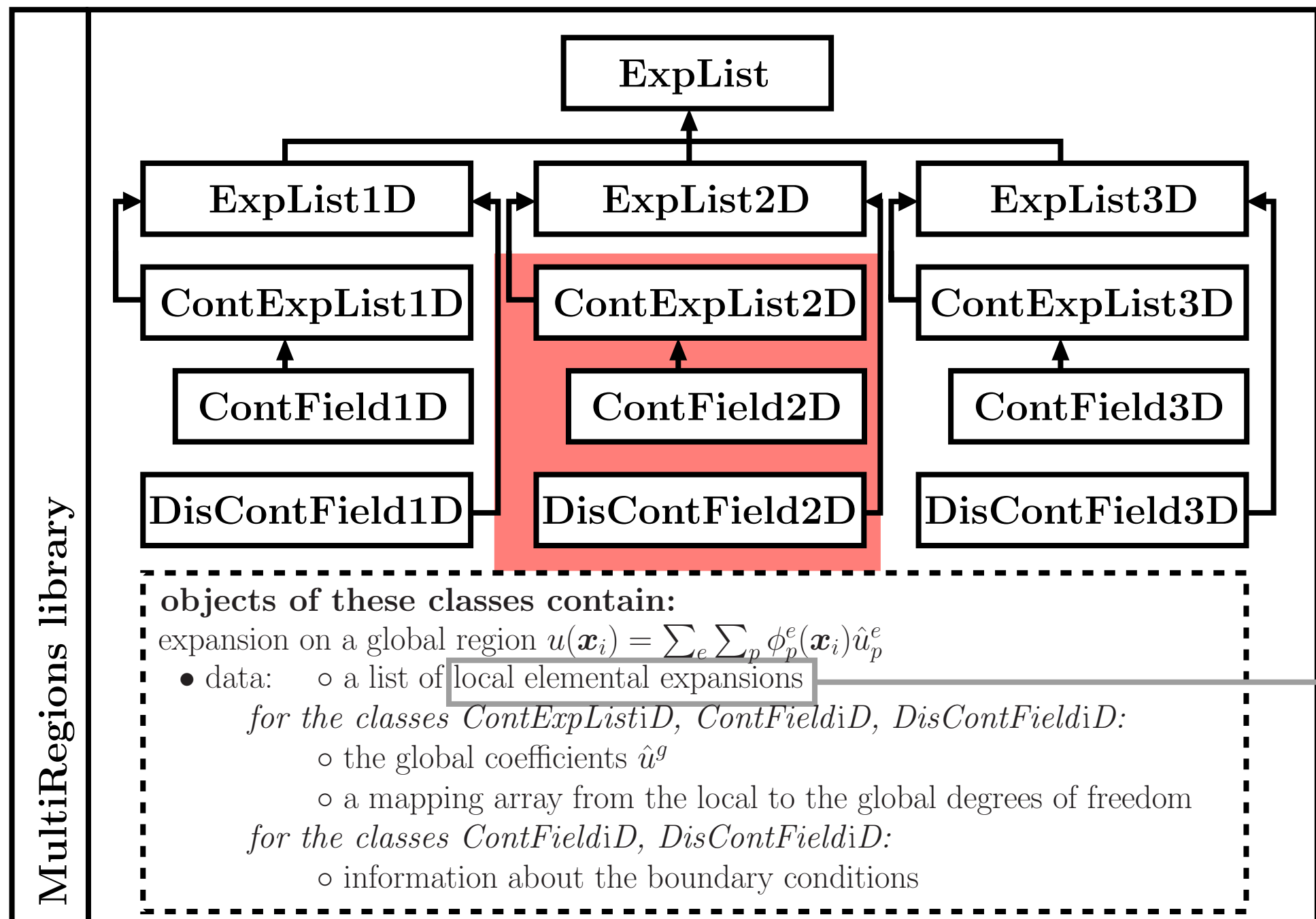
Nektar++: Helmholtz solver



Nektar++: Helmholtz solver



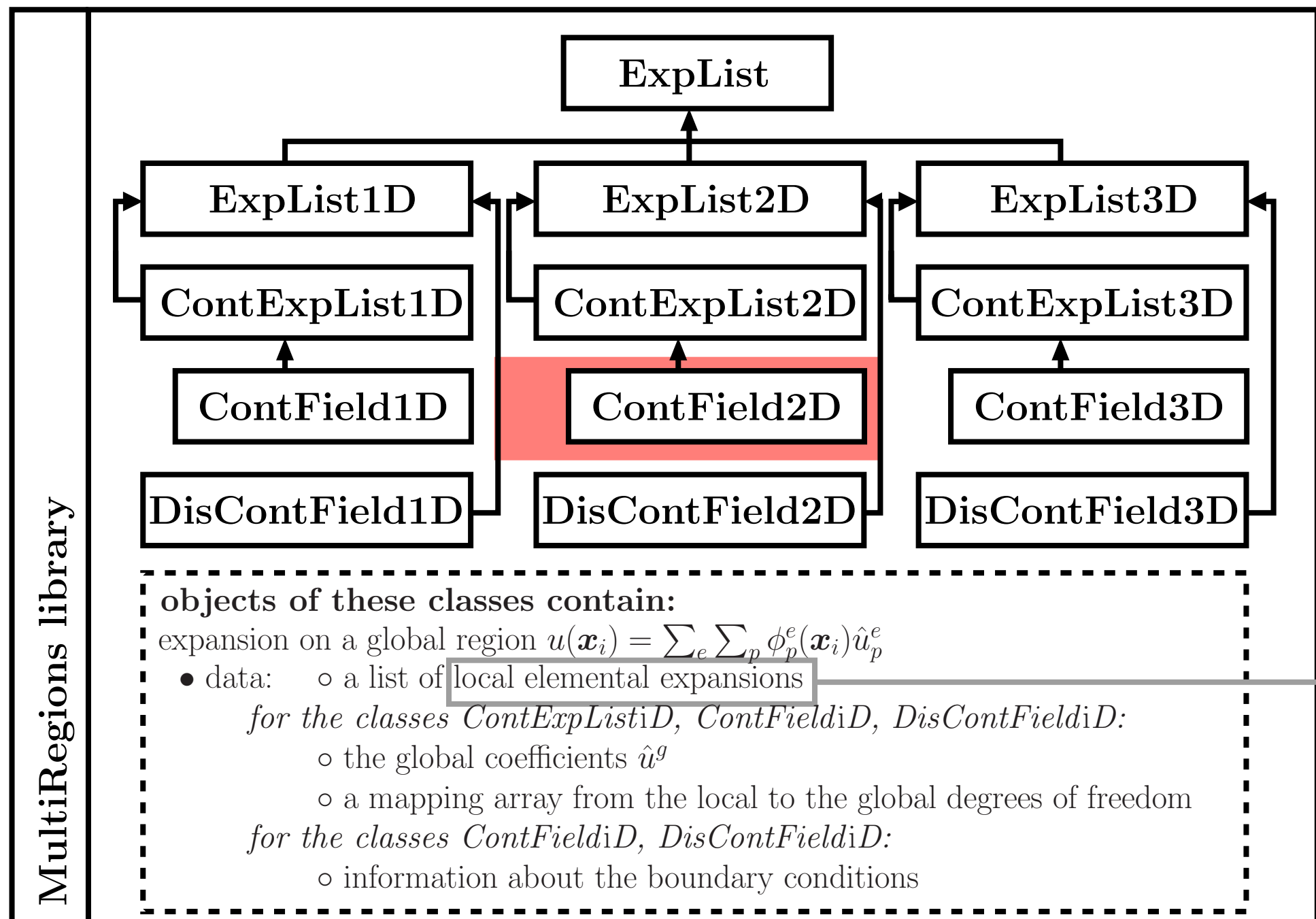
Nektar++: Helmholtz solver



Nektar++



Nektar++: Helmholtz solver



Nektar++: output

- Gmsh
- Tecplot

Assignment

1 element , $P = 2$ (NUMMODES= $P + 1 = 3$)
--

h –refinement	P –refinement
-----------------	-----------------

4 elements, $P = 2$	1 element , $P = 4$
---------------------	---------------------

9 elements, $P = 2$	1 element , $P = 6$
---------------------	---------------------

16 elements, $P = 2$	1 element , $P = 8$
----------------------	---------------------

Nektar++: classes

- class
 - expanded concept of a data structure
 - data
 - functions

quadrilateral standard
expansion



abstraction

$$u^\delta(\xi_{1_i}, \xi_{2_i}) = \sum_{n=0}^{N-1} \hat{u}_n \phi_n(\xi_{1_i}, \xi_{2_i})$$

Class: StdQuadExp

- data members

- m_ncoeffs
- m_coeffs
- m_base
- m_phys
- ...

- member functions

- BwdTrans()
- ...

Nektar++: classes

- inheritance

Class: StdTriExp

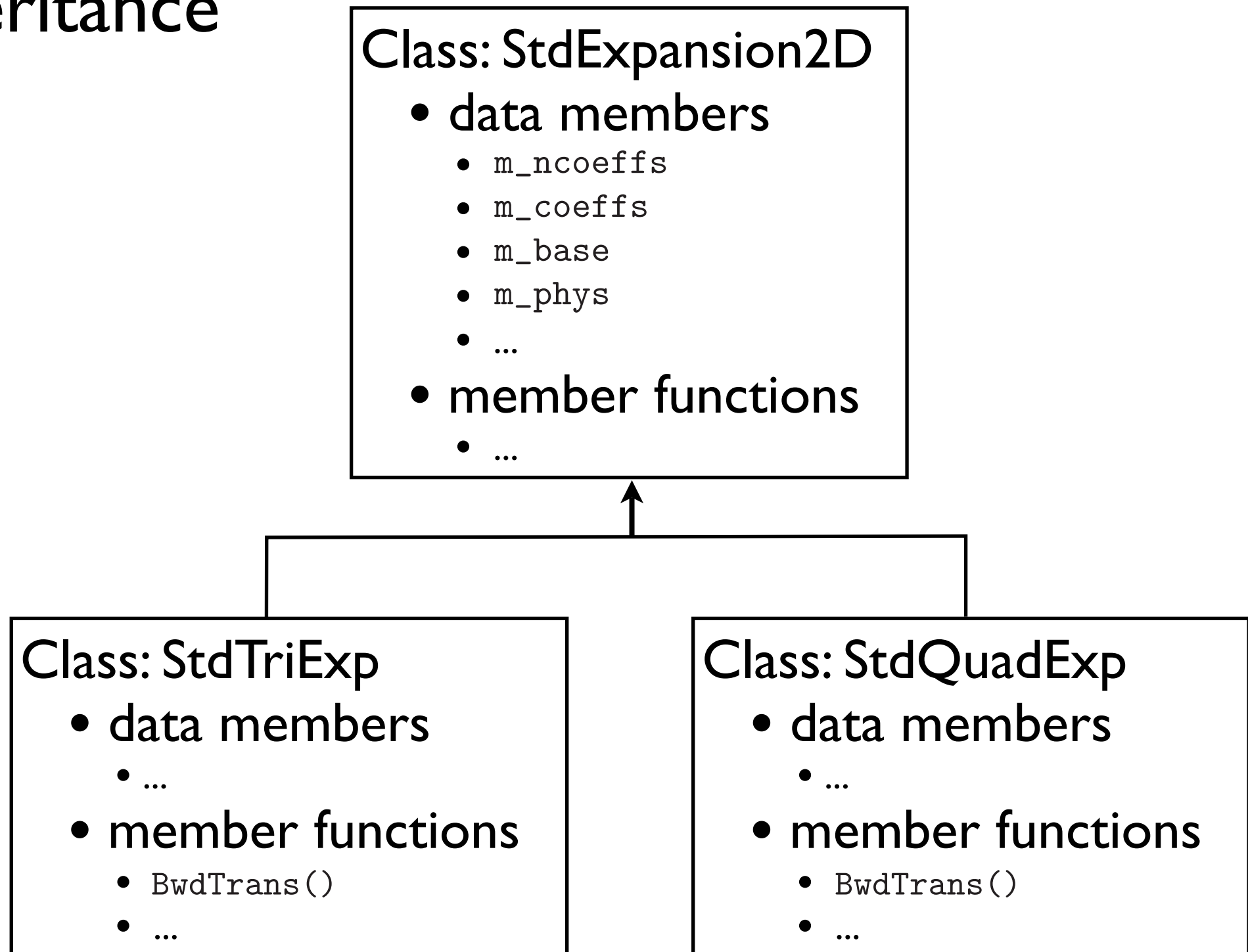
- data members
 - `m_ncoeffs`
 - `m_coeffs`
 - `m_base`
 - `m_phys`
 - ...
- member functions
 - `BwdTrans()`
 - ...

Class: StdQuadExp

- data members
 - `m_ncoeffs`
 - `m_coeffs`
 - `m_base`
 - `m_phys`
 - ...
- member functions
 - `BwdTrans()`
 - ...

Nektar++: classes

- inheritance



Nektar++: classes

