Unified Solids

-   Creation of a new class   -
UMultipleUnion

Tuesday, June 28th 2011

General overview (1)

1) **DistToIn** → calculates the *distance* from a *point located inside the solid* to the surface of the latter

2) **DistToOut** → idem, but for a *point located outside the solid*

3) **Safety** → computes the *closest distance* from a given *point* to the considered *solid*.
(2 versions: SafetyFromInside/Outside)

4) **Extent** → determines the *extension of the solid* in the *form of a box*

5) **Normal** → computes the *normal* to the closest surface from the considered point (+ direction)
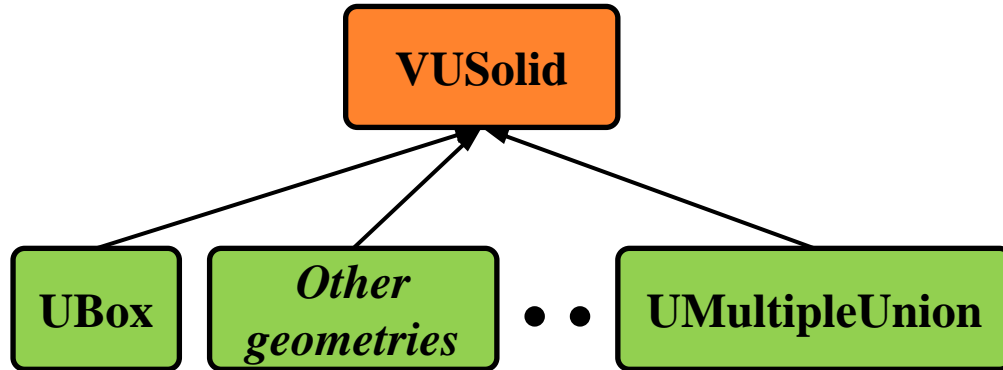
- **DistFromOutside**

- **DistFromInside**

- **Safety**

- **ComputeBBox**

- **ComputeNormal**

General overview (2)

1) **DistToIn** → calculates the *distance* from a *point located inside the solid* to the surface of the latter
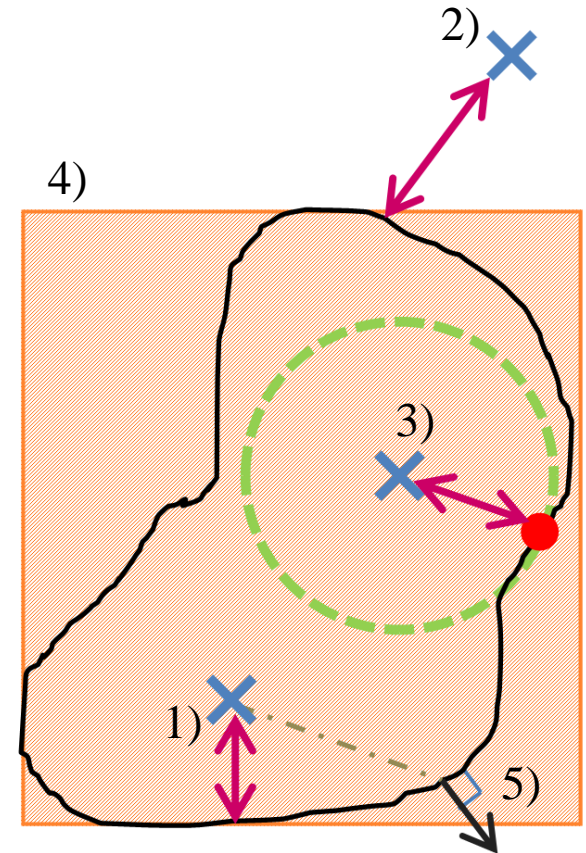
2) **DistToOut** → idem, but for a *point located outside the solid*

3) **Safety** → computes the *closest distance* from a given *point* to the considered *solid*.
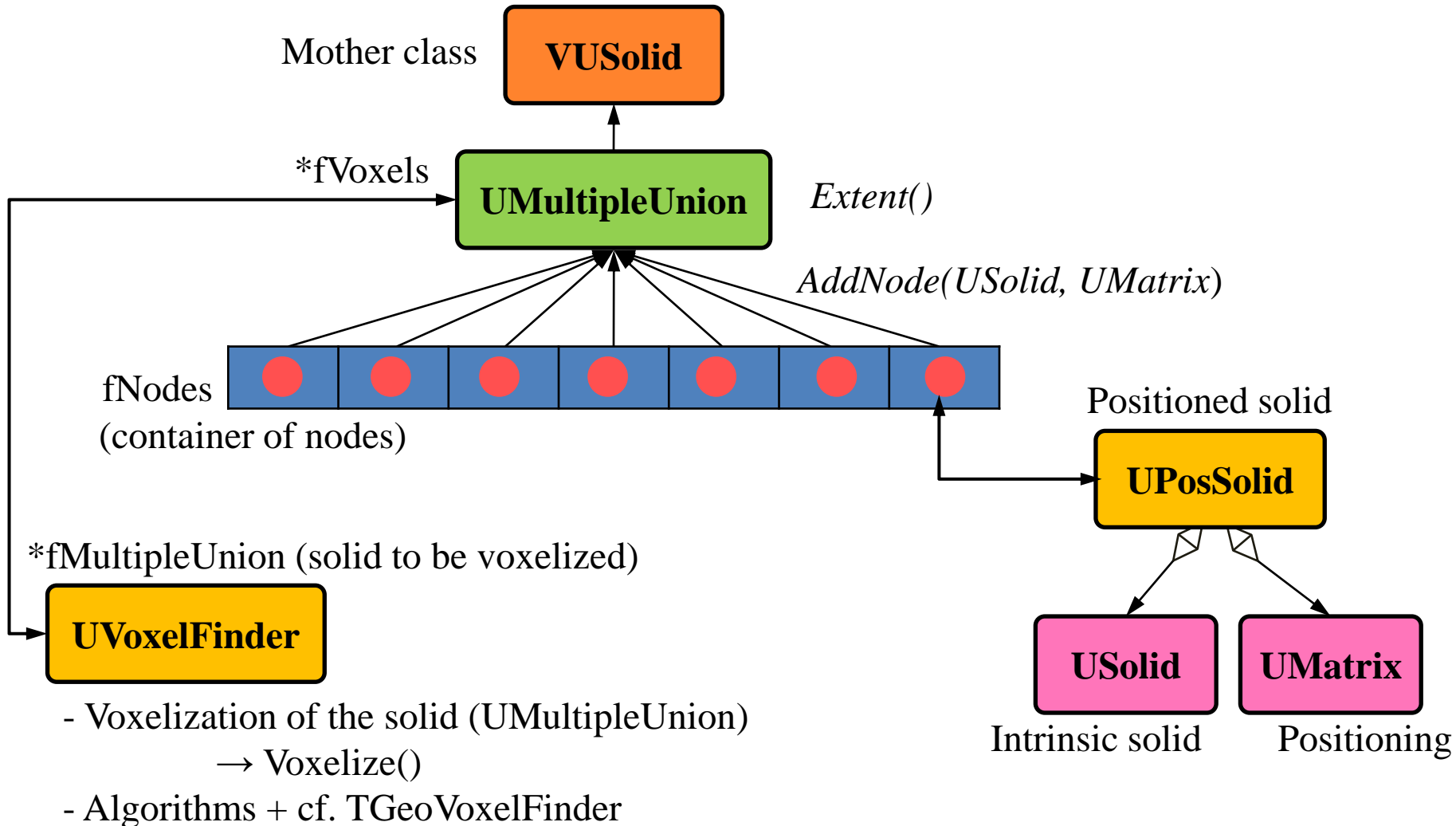(2 versions: SafetyFromInside/Outside)

4) **Extent** → determines the *extension of the solid* in the *form of a box*

5) **Normal** → computes the *normal* to the closest surface from the considered point (+ direction)
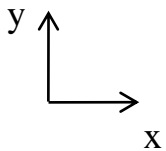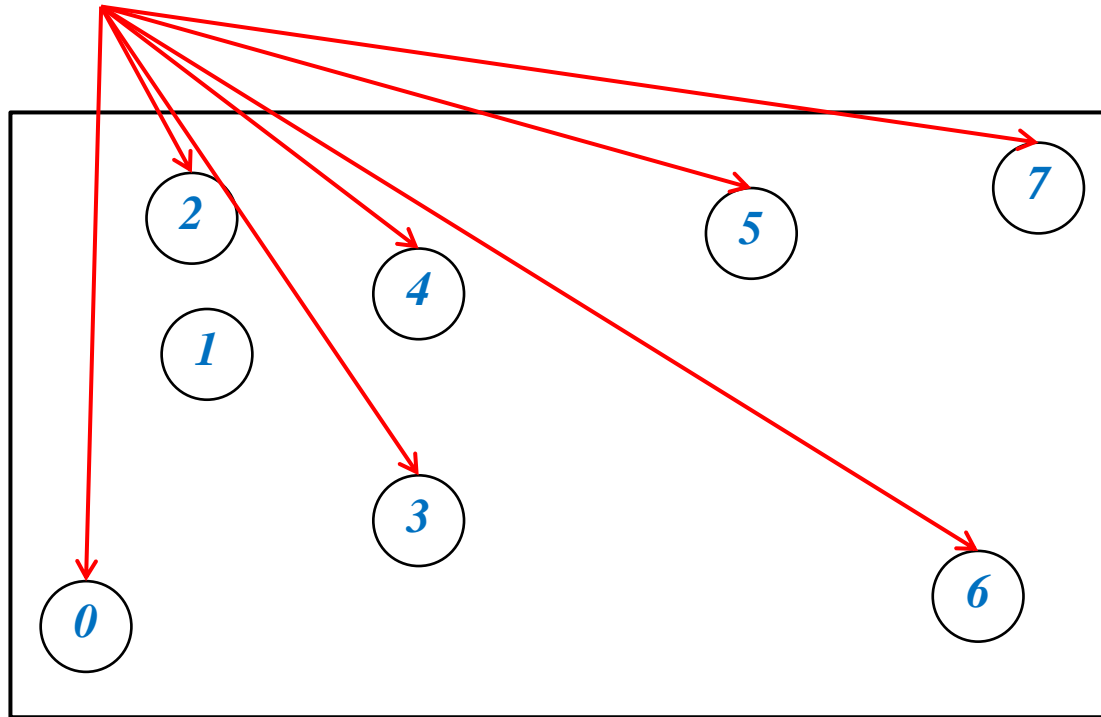
UMultipleUnion class

- Synoptic diagram -

Mother class **VUSolid**

*fVoxels **UMultipleUnion** *Extent( )*

AddNode(USolid, UMatrix)

fNodes
(container of nodes)

Positioned solid

UPosSolid

*fMultipleUnion (solid to be voxelized)

UVoxelFinder

- Voxelization of the solid (UMultipleUnion)
    → Voxelize()
- Algorithms + cf. TGeoVoxelFinder

USolid

Intrinsic solid

UMatrix

Positioning
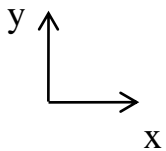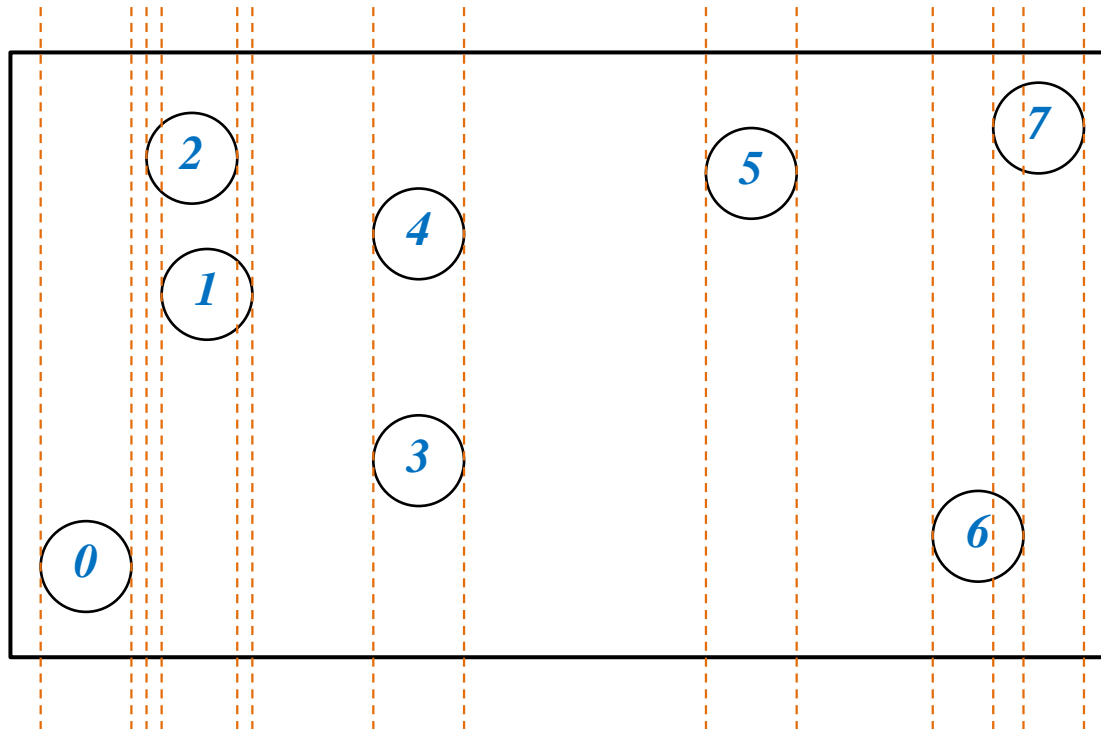
4

Voxelization (1)

A. Determination of the boundaries of each node

MultiUnionInstance → GetNode(id)

A. Determination of the boundaries of each node
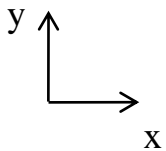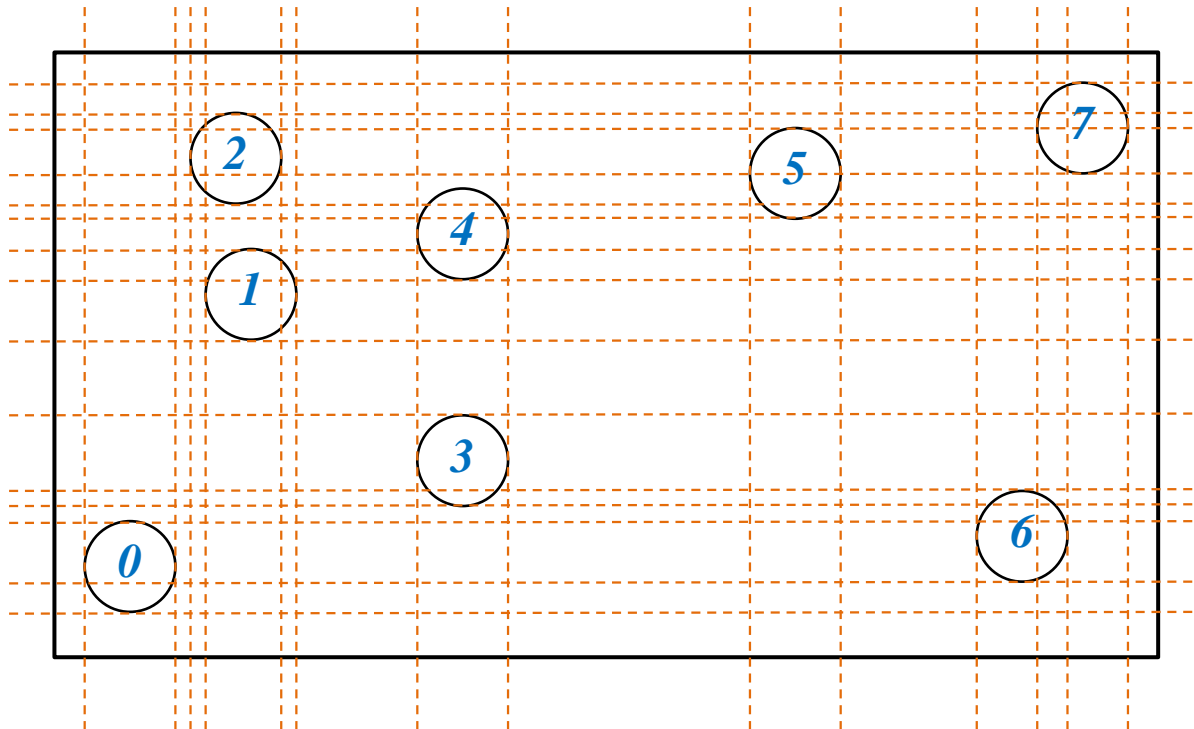
→ along x axis

Voxelization (3)

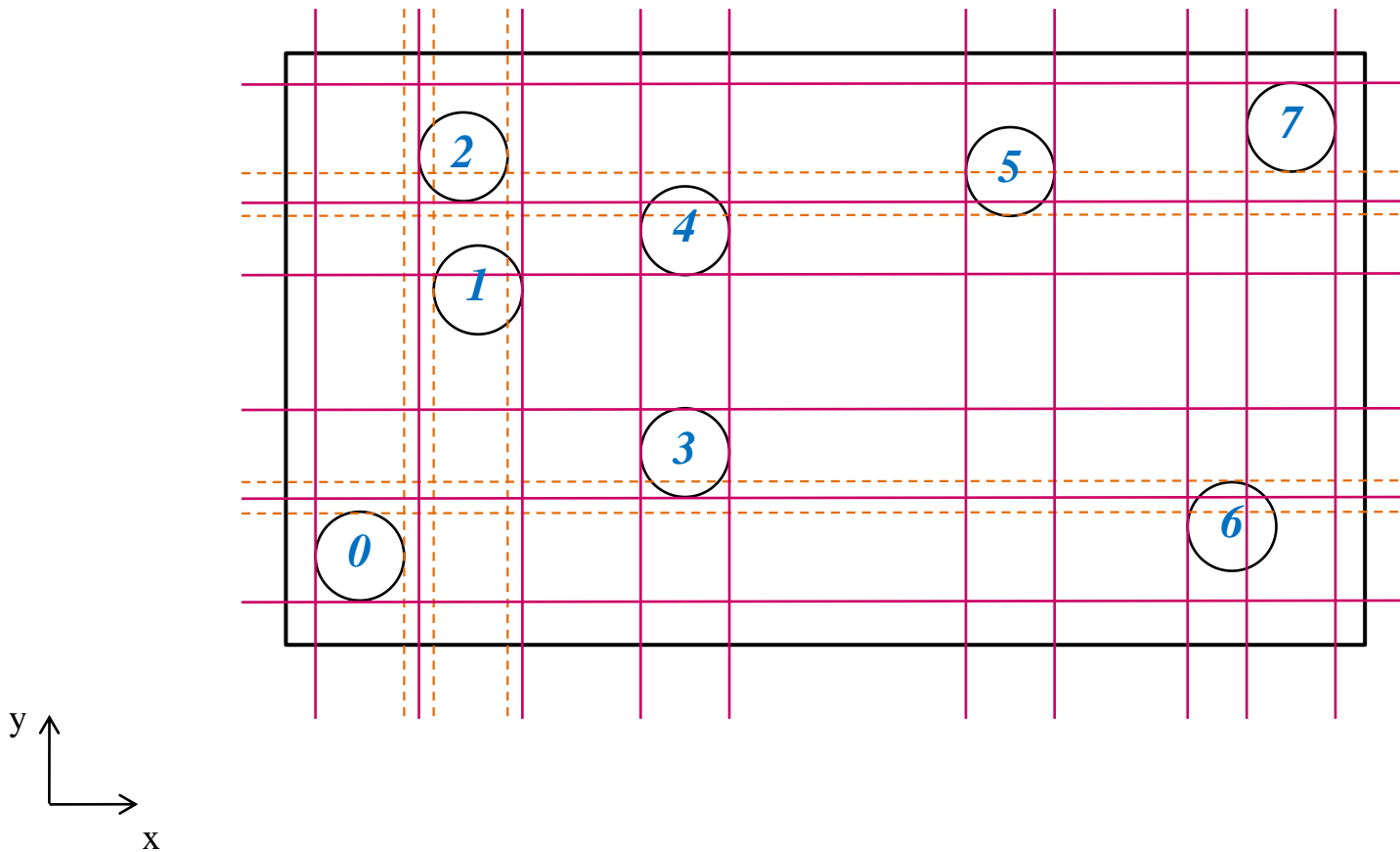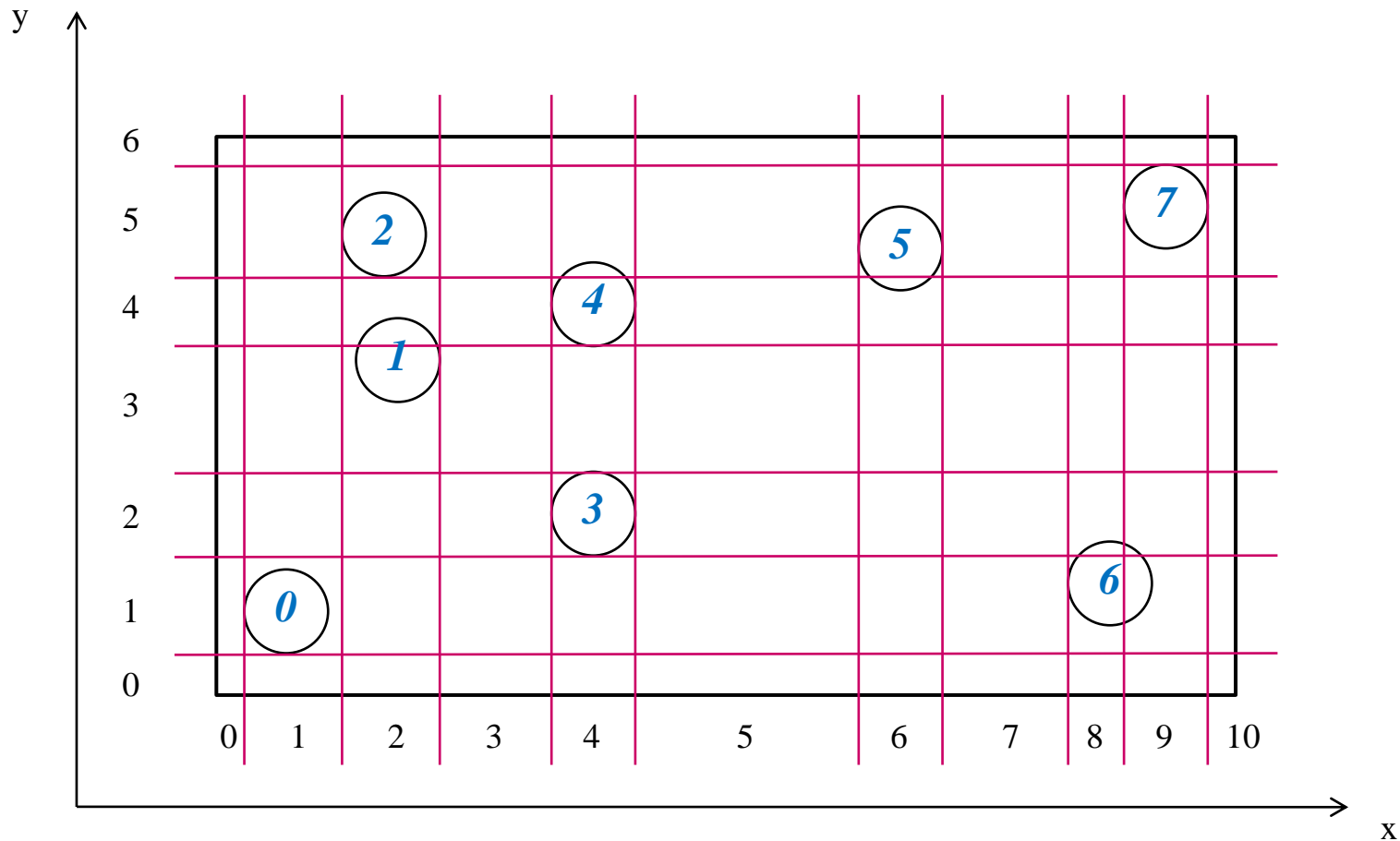*A. Determination of the boundaries of each node*

→ along x axis
→ along y axis

B. Deletion of too close boundaries

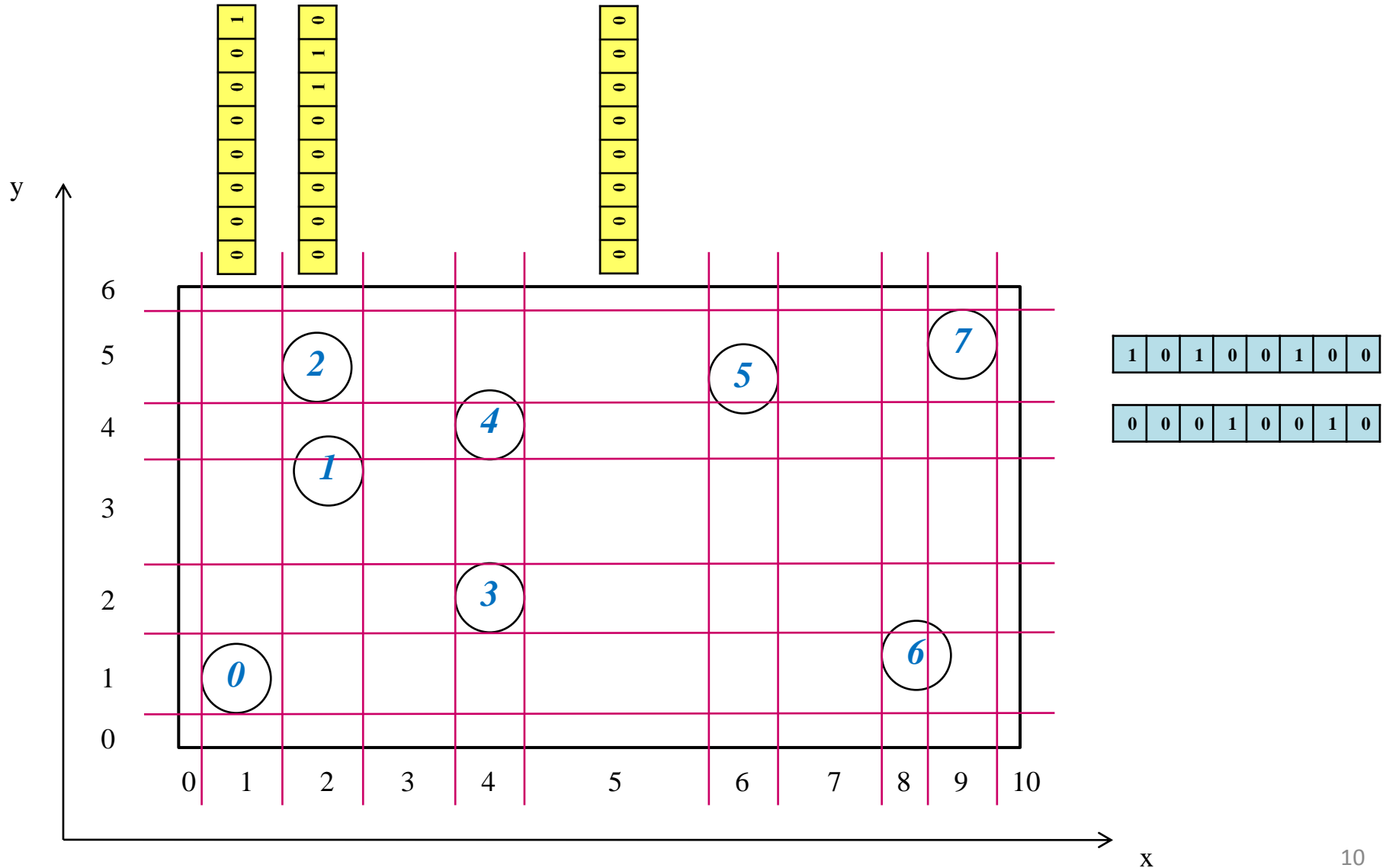→ Suppression when the distance between two boundaries is inferior to a preset coefficient δ

B. Deletion of too close boundaries

C. Memorization of the nodes located in each voxel

Voxelization (7)

D. Summary

Determination of the limits
associated to each node
along x, y and z

↓

Sorting and selection (if
need be) of the boundaries

↓

Memorization of the nodes
present in each voxel
previously determined

↓

Determination of the nodes
located in an adjacent
voxel (propagation)