

Approche pratique/intégrée expérimental/simulation en aérodynamique

Maillage

Nicolas Doué
nicolas.doue@isae.fr



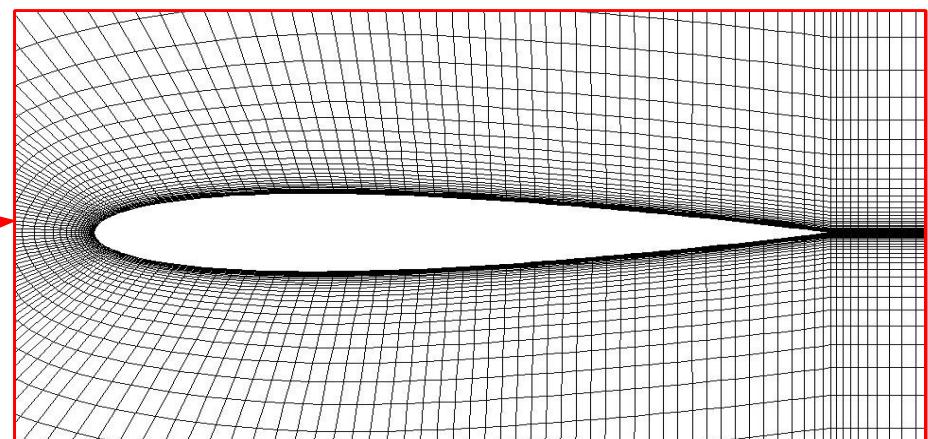
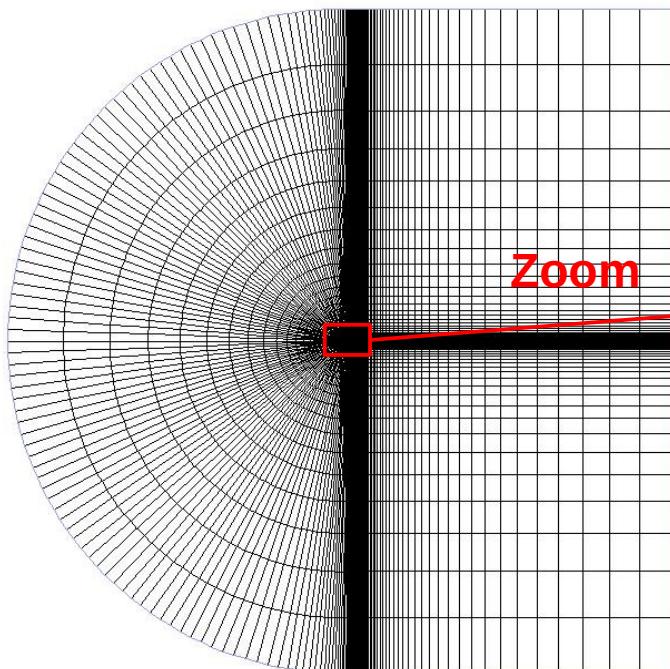
Mars 2017

Mesh & Computation Domain

Computation domain

- A **Mesh** (or Grid) is a spatial discretization of a continuous domain. In CFD, this continuous domain, called **computation domain**, is the volume in which the flow is computed.

Example of mesh for NACA0012 profile



- Solids do not need to be meshed (unless we need a thermal analysis inside the solids).

Boundary conditions

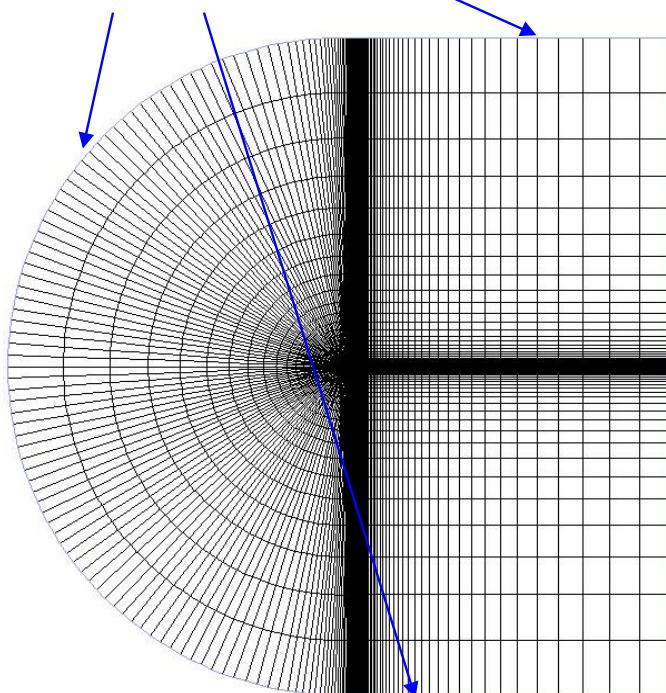
- A Mesh has borders called « **Boundary Conditions** ».

In most cases, we need at least :

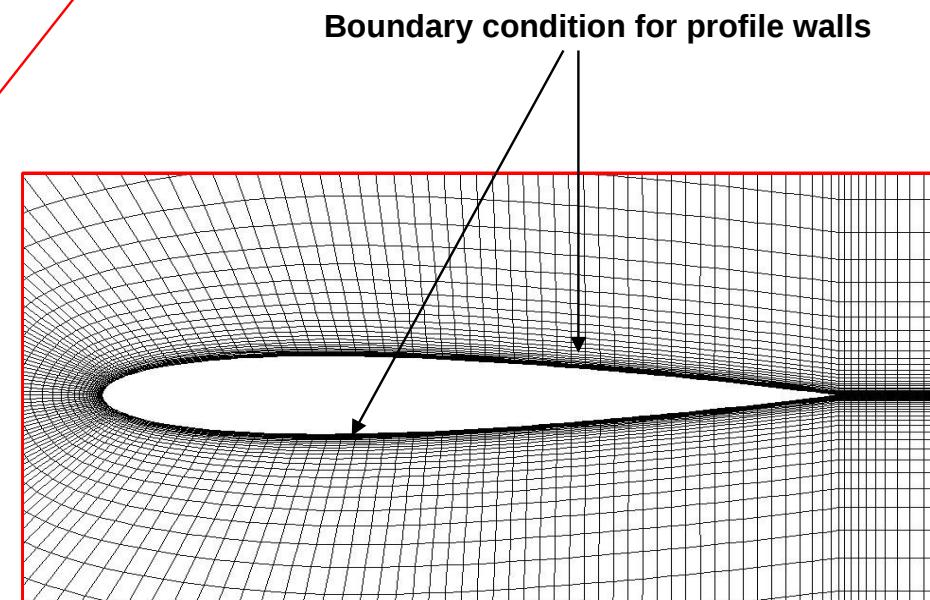
- one inlet
- one outlet
- walls

Example with the NACA0012 profile

Boundary condition for domain inlet



Boundary condition for domain outlet



Boundary condition for profile walls

Size of the computation domain (1/2)

In CFD softwares, uniform fields (velocity magnitude and direction, pressure ...) are usually defined for the boundary conditions.

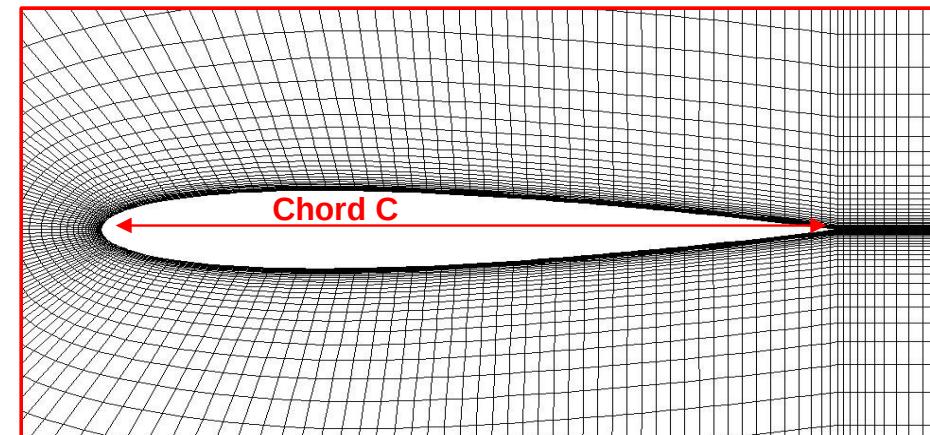
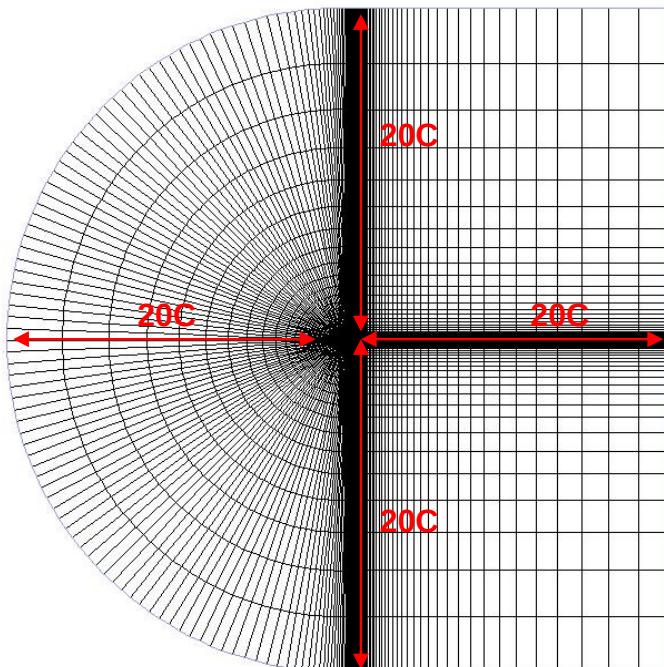
If the boundary conditions are close to the phenomenon we want to analyse, these uniform fields can have huge influences on the final results and lead to wrong results.

This is the reason why the boundary conditions must be moved far away from the studied phenomenon.

For external flow, the boundary conditions are moved at 10 characteristic lenght, at least, from the studied object.

Example with the NACA0012 profile

Boundary conditions are placed to 20 chords from the profile



Size of the computation domain (2/2)

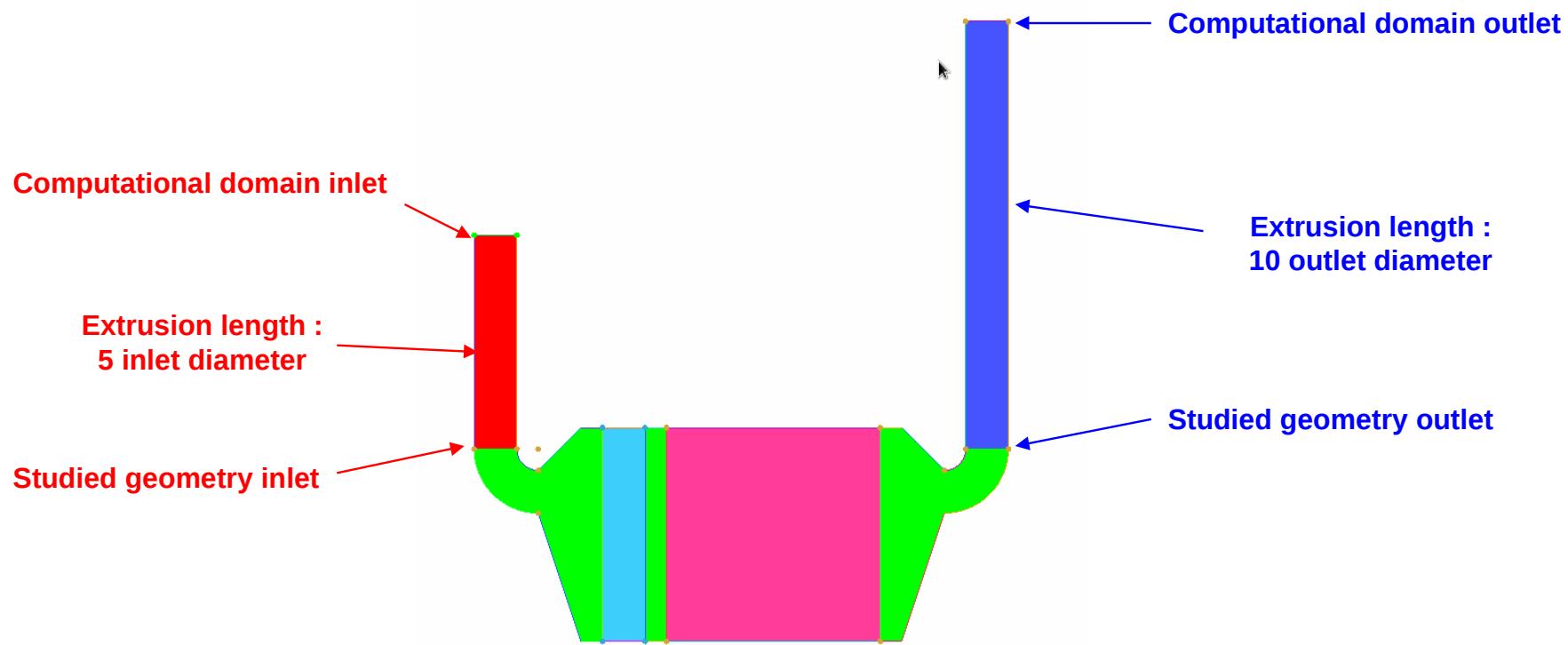
For internal flow, the inlets and outlets of the studied system are very often extruded to decrease the influences of the uniform fields.

The extrusion lenght are commonly :

- for inlets, at least 5 maximum lenght of the inlet
- for outlets, at least 10 maximum lenght of the outlet

Example with catalytic converter

Boundary conditions are placed at 5 inlet diameters for inlet and 10 outlet diameters for outlet



Cells

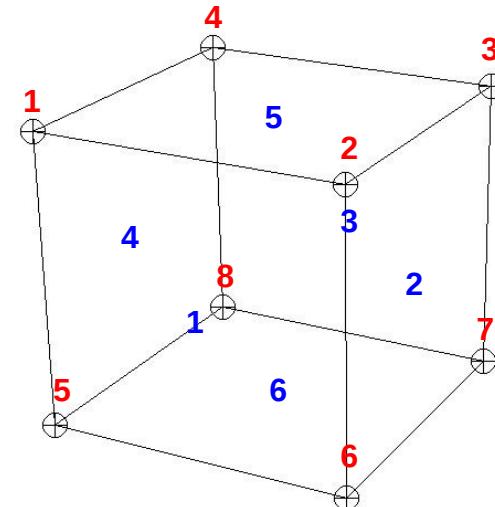
Generalities about Cells (1/3)

A Mesh is composed of :

- nodes
- faces
- cells (or elements)

For example, a hexahedral cell has :

- 8 **nodes**
- 6 **faces**



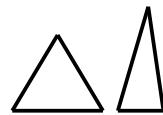
There are different kinds of cells :

- tri, quad or polygon cells for 2D meshes
- tetrahedral, pyramidal, prism, hexahedral or polyhedral cells for 3D meshes

Examples of 2D cells

Tri:

- 3 nodes
- 3 faces



Examples

Quad:

- 4 nodes
- 4 faces



Examples

Polygon:

- at least 5 nodes
- at least 5 faces



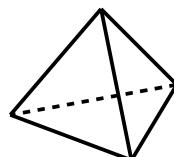
Examples

Generalities about Cells (2/3)

Examples of 3D cells

Tetrahedral :

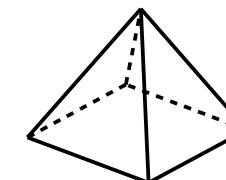
- 4 nodes
- 4 faces



Example

Pyramidal :

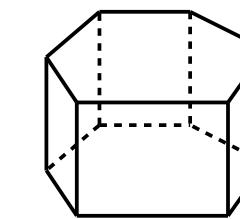
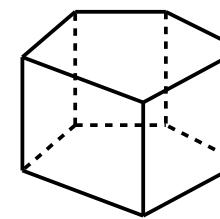
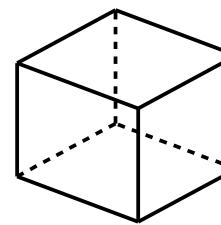
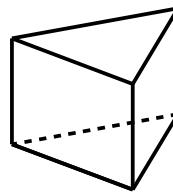
- 5 nodes
- 5 faces



Example

Prism :

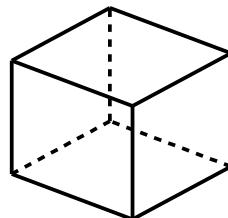
- at least 6 nodes
- at least 5 faces



Examples

Hexahedral :

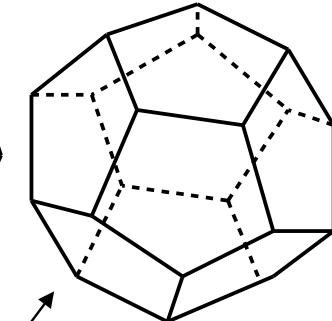
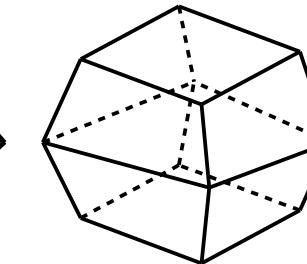
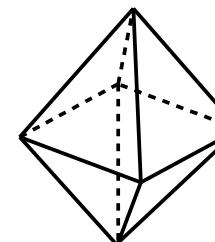
- 8 nodes
- 6 faces



Example

Polyhedral :

- at least nodes
- at least 7 faces

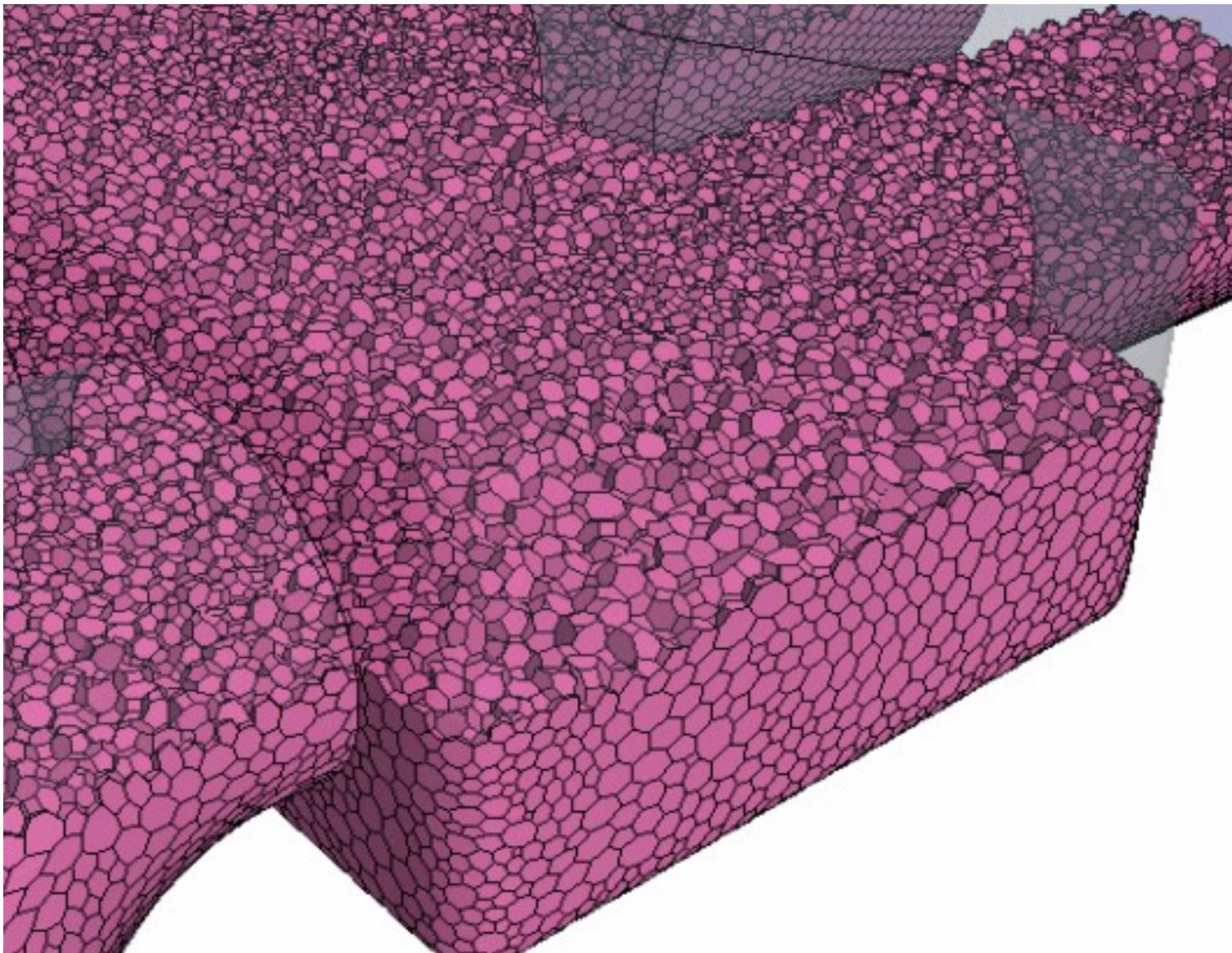


Examples

Common polyhedral cell

Generalities about Cells (3/3)

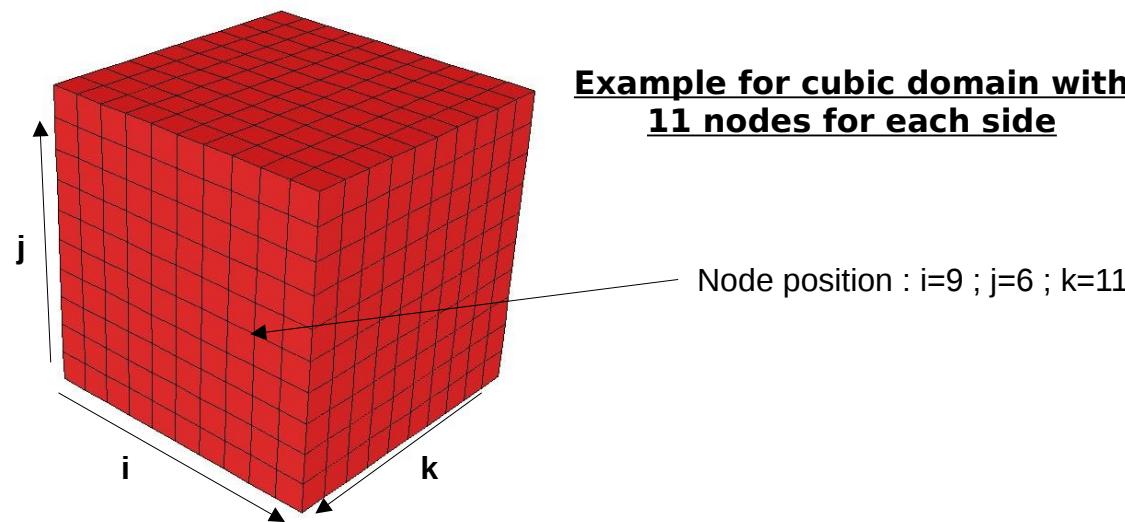
Example of polyhedral mesh with Star CCM+



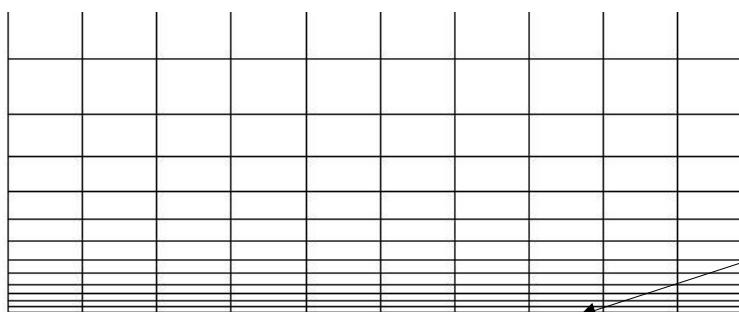
Structured & Unstructured Meshes

12 Structured meshes

- A **structured mesh** has node whose location can be defined by indices. The number of indices is equal to the mesh dimension.



- Cell size is not always uniform. A mesh can be locally **refined** (decrease in cell size) for a better discretization of gradients.



Example of structured mesh with refinement near the wall for the discretization of velocity gradients in the boundary layer

No-slip wall

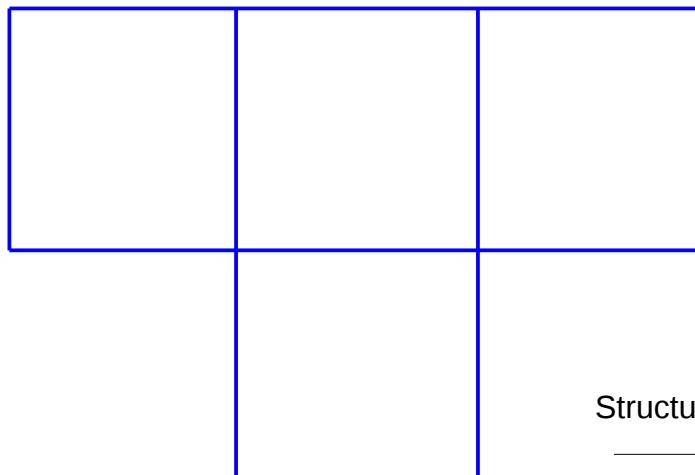
Block structured meshes

To build meshes for computation domains with more complex shapes, we can use several blocks.

A **block structured Mesh** is built by several blocks with structured mesh.

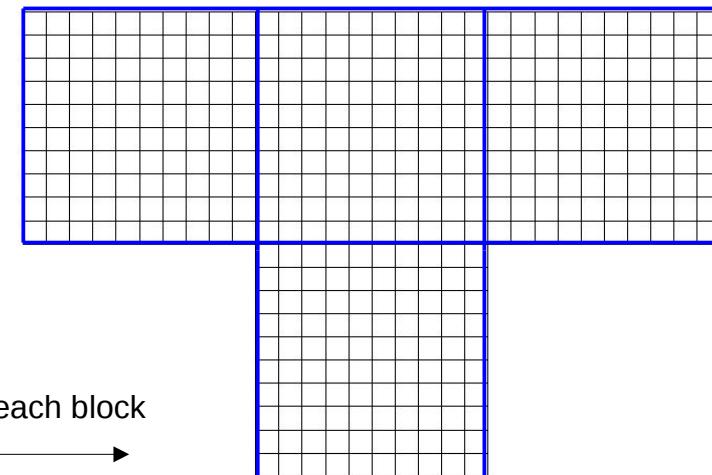
Example of computation domain built with 4 blocks.

A structured mesh for each block provides a block structured mesh for the whole domain.



4 blocks

Structured mesh for each block

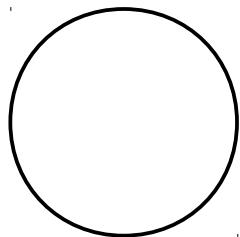


Block structured mesh

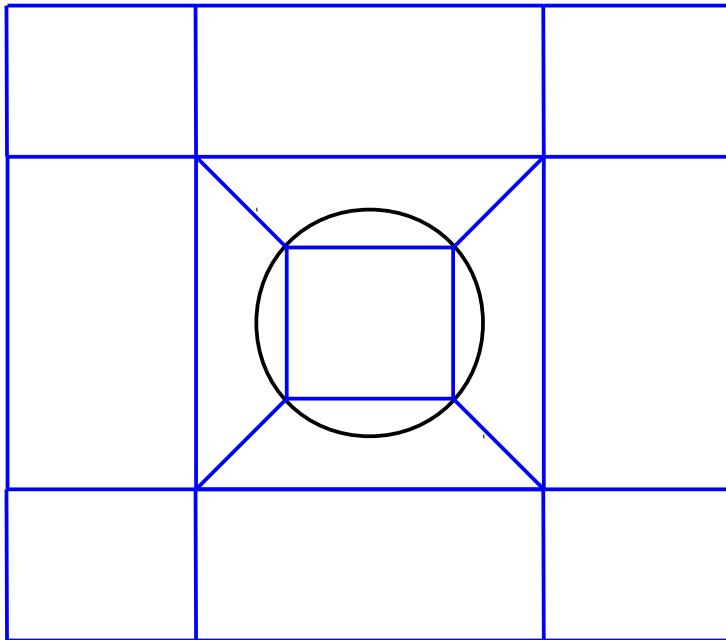
Block structured meshes : O-Mesh

One typical block structured mesh is the O-Mesh. Blocks are placed around the geometry and take the shape of a « 0 ».

Geometry



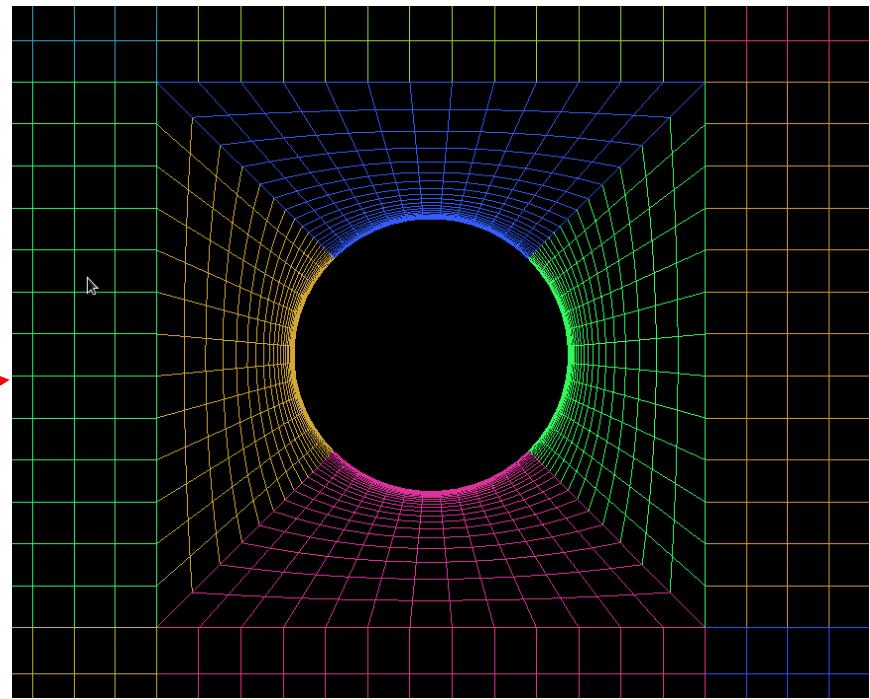
Blocking



Example of O-Mesh around a cylinder

Blocks are placed around the geometry like a « O »

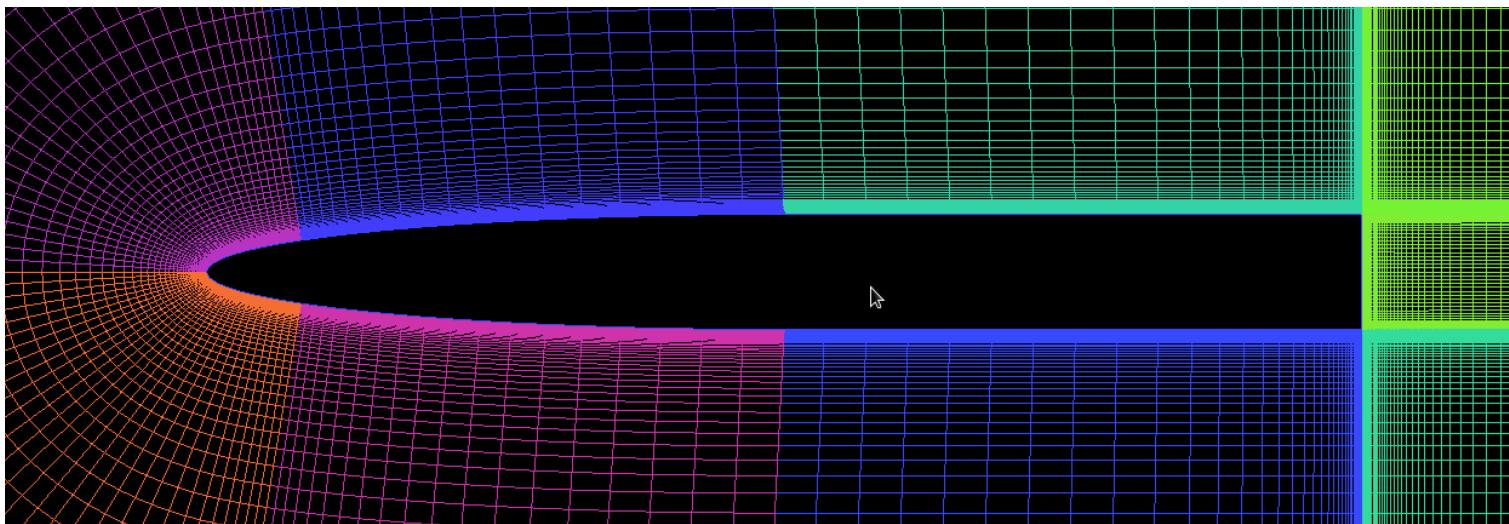
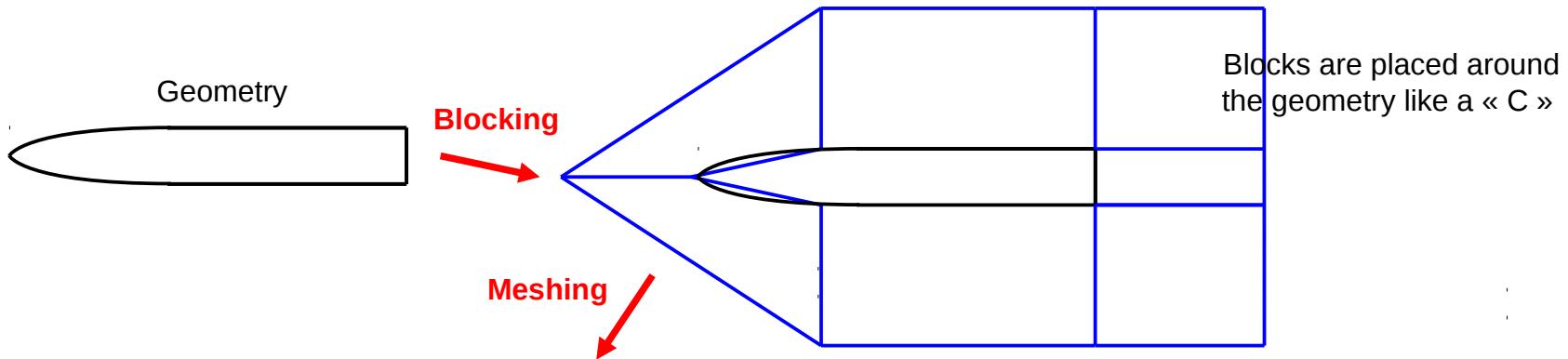
Meshing



Block structured meshes : C-Mesh

Another typical block structured mesh is the C-Mesh, often used for profiles or blunt base bluff bodies. Blocks take the shape of a « C ».

Example of C-Mesh



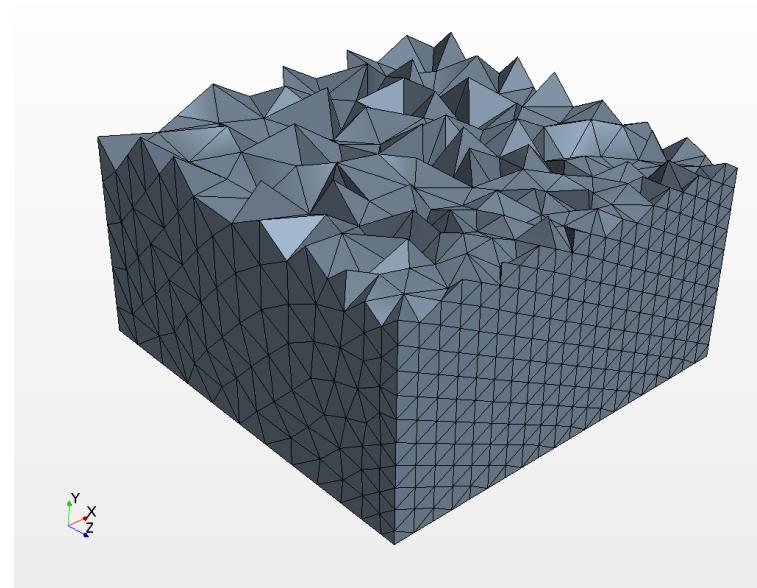
Unstructured meshes (1/2)

An unstructured mesh is a discretization of the domain by cells with irregular connectivities.

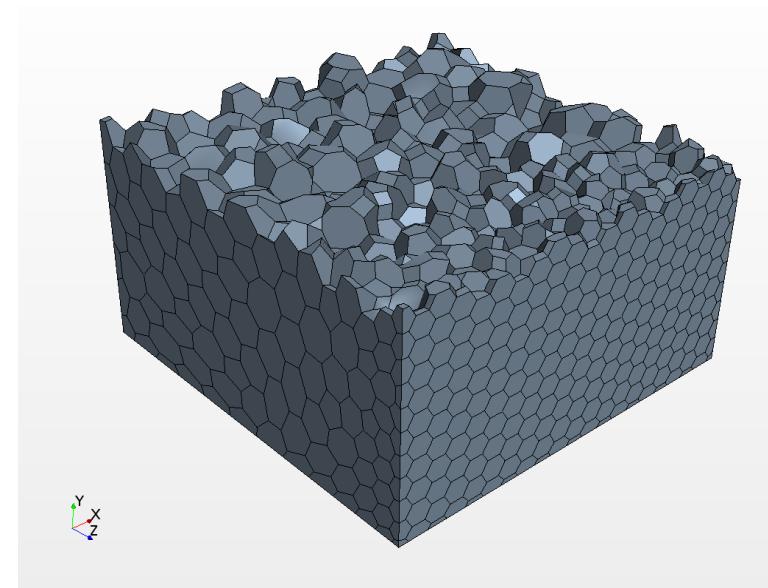
Unlike the structured mesh, which uses quad (for 2D mesh) or hexahedral (for 3D mesh) cells, the unstructured mesh can use all type of cells:

- tri, quad or polygon cells for 2D mesh
- tetrahedral, pyramidal, prism, hexahedral or polyhedral cells for 3D mesh.

Tetrahedral cells

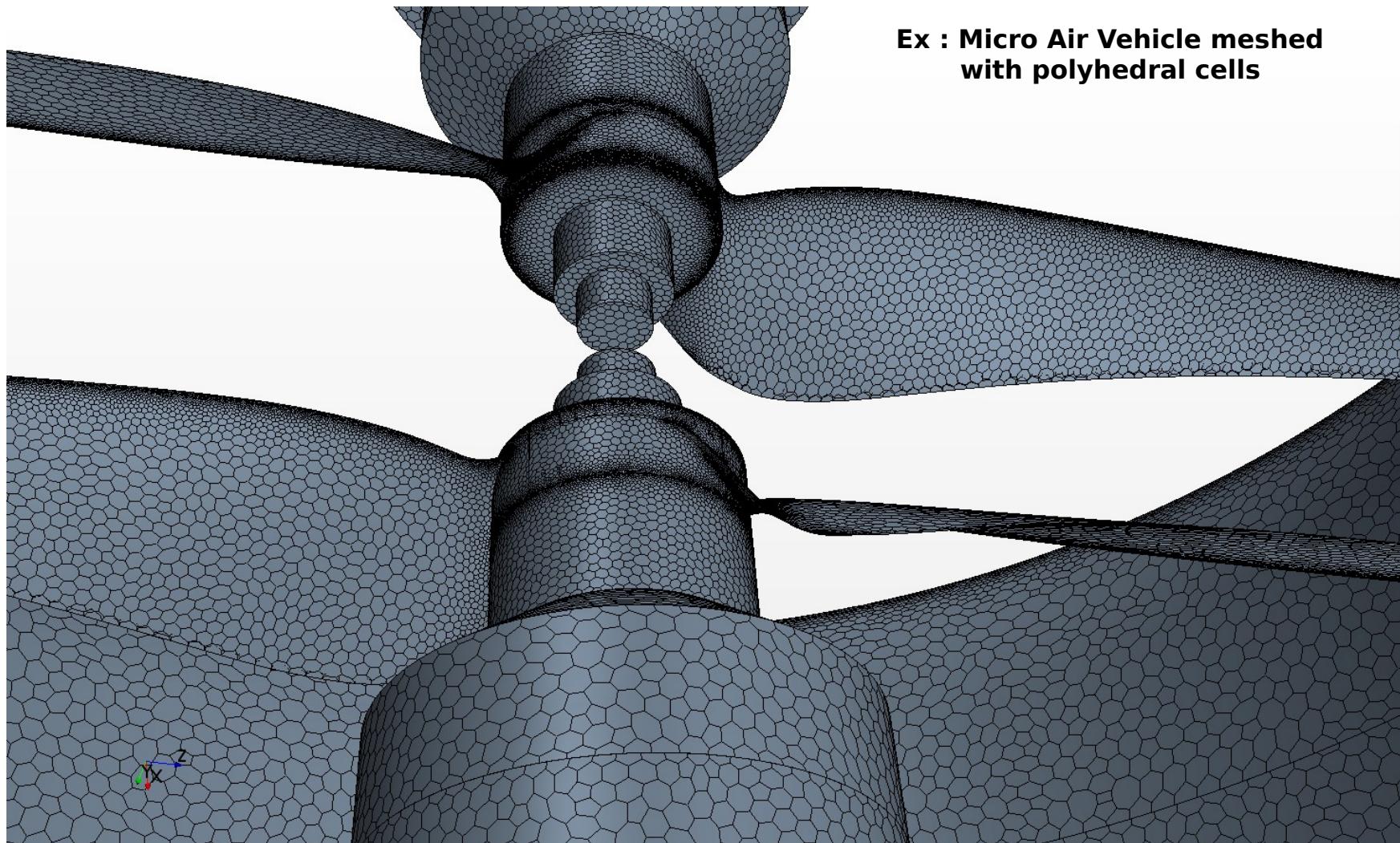


Polyhedral cells



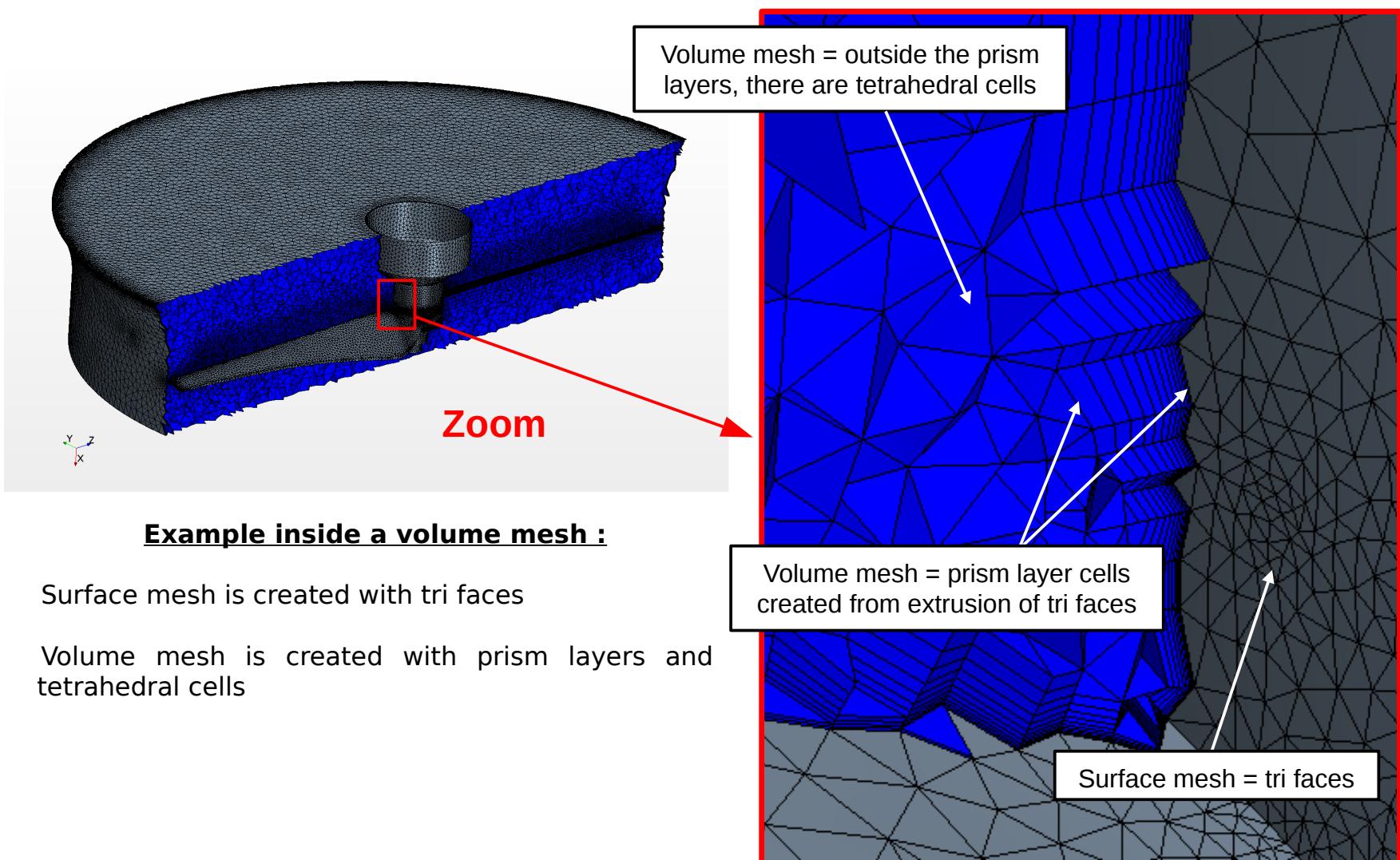
Unstructured meshes (2/2)

Unstructured meshes are often generated for complex shape geometries, which are very long or impossible to mesh correctly with block structured meshes.



Unstructured meshes : prism layers

Unstructured meshes need **prism layers** on walls to discretize the boundary layer. Prism layers are prismatic cells extruded from the surface mesh.

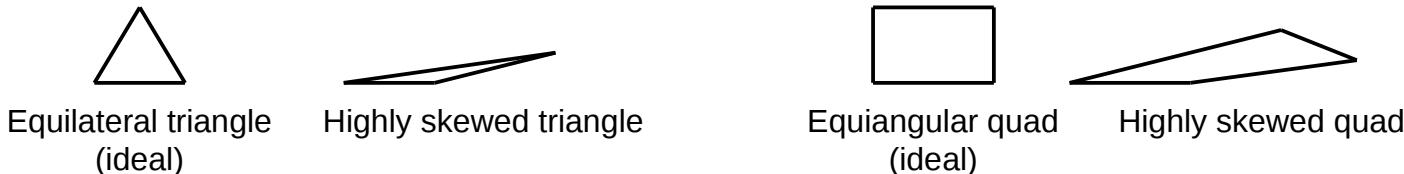


Mesh Quality Criteria

Skewness (Theory)

Skewness is one of the primary quality measures for a mesh.

Skewness determines how close to ideal (equilateral for triangles or equiangular for quads) a cell is.



The equation of the Normalized Equiangular Skewness is :

$$\max \left[\frac{\theta_{\max} - \theta_e}{180 - \theta_e}, \frac{\theta_e - \theta_{\min}}{\theta_e} \right]$$

where

θ_{\max} = largest angle in the cell

θ_{\min} = smallest angle in the cell

θ_e = angle for an equiangular cell (60° for a triangle, 90° for a square)

Skewness	Cell quality
0	ideal
$>0 - 0.25$	excellent
$0.25 - 0.5$	good
$0.5 - 0.75$	fair
$0.75 - 0.9$	poor
$0.9 - <1$	bad (sliver)
1	degenerate

Equation for Skewness in Ansys
ICEM software is :
 $1 - \max(\dots)$

Thus :
1 is ideal cell
0 is degenerate cell

Skewness (Practical 1/2)

With unstructured mesh for complex shape geometries, we can have highly skewed cells. The main reason is the cells have not enough space in very closed volumes.

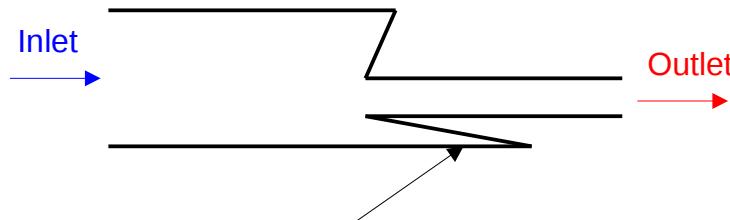
If we have one cell with a skewness > 0.9 , the computation convergence will be very difficult.

Convergence is sometimes possible with only one cell with a skewness of 0.91 or 0.92, but can lead to local non-physical results (example : local velocity of 10,000 m/s).

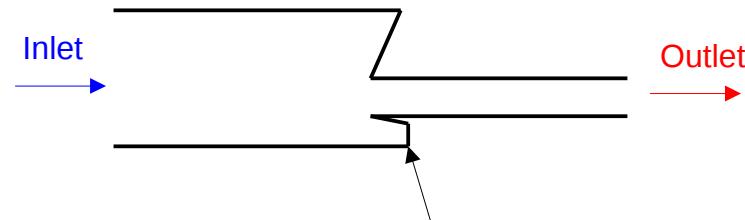
The mesh can often be improved by :

- decreasing the mesh size
- decreasing or (deleting) the prism layer thickness
- changing locally the geometry (if there is no influence on the flow)

Example of geometry change :



Difficult to mesh low skewed cells in this part of the domain.
Especially with the presence of prism layers.



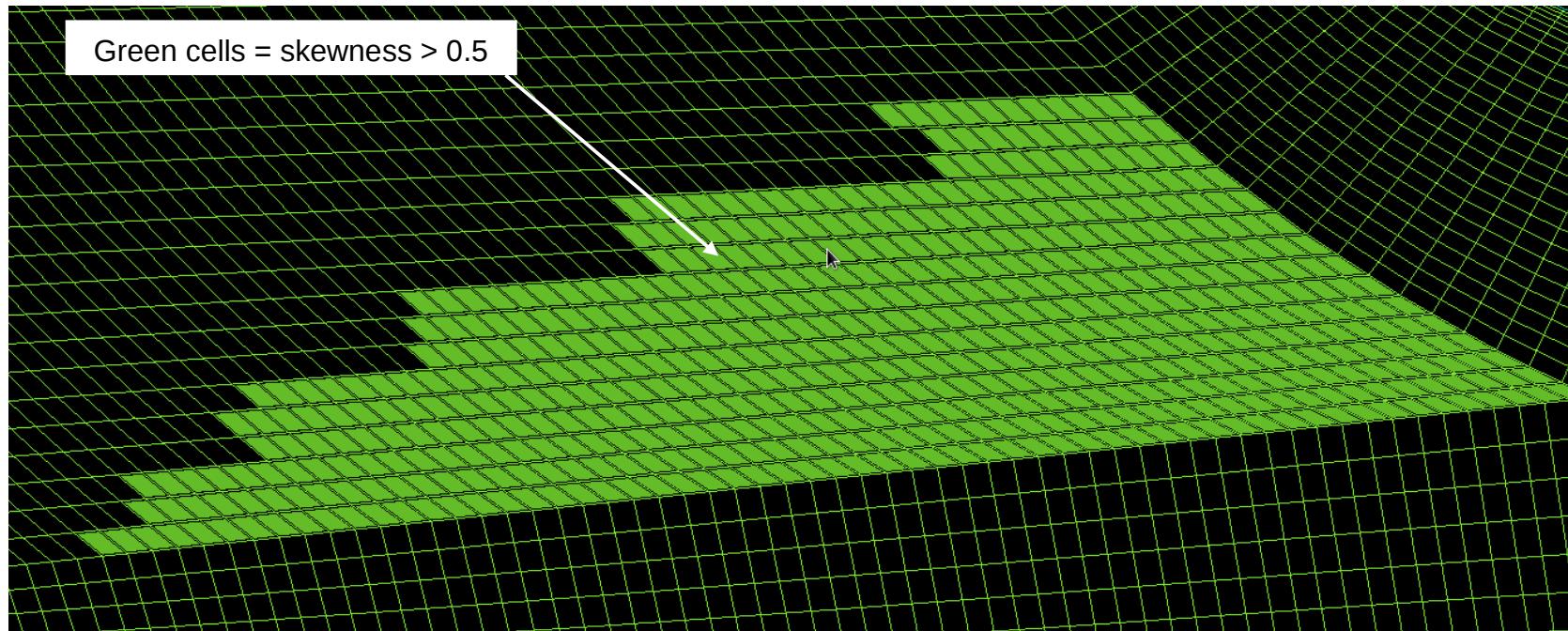
The geometry is cut without changing the flow

Skewness (Practical 2/2)

With structured mesh, it is advised to have skewness < 0.5 for all cells.

Only one cell with such a skewness is not problematic. But in blocks used for structured meshes, a cell has approximately the same skewness as its neighbours. Thus we can have a lot of cells with a fair skewness, that can lead to divergence.

Example of high skewed cells in a block structured mesh :



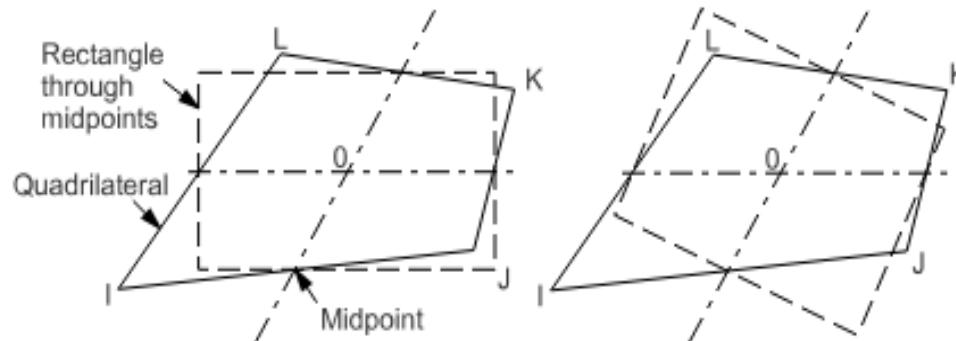
To have skewness < 0.5 , $45^\circ < \text{cell angles} < 135^\circ$

Aspect ratio (Theory)

Aspect ratio is another primary quality measure for a mesh.

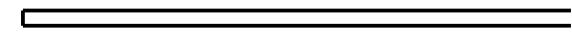
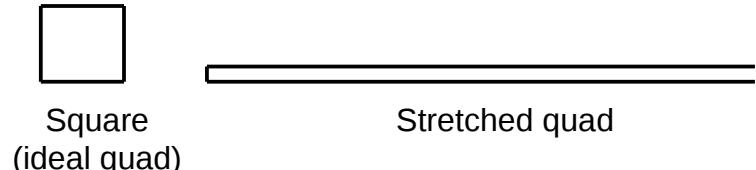
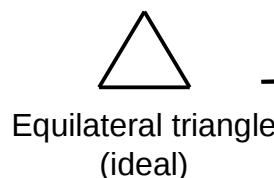
Aspect ratio determines how close to ideal (equilateral for triangles or square for quads) cell sides are.

For a IJKL quad cell, rectangles passing through the cell edge midpoints are constructed. The aspect ratio of the IJKL quad cell is the ratio of a longer side to a shorter side of whichever rectangle is most stretched.



The aspect ratio for a triangle is not described here, because we obtain a skewed triangle before a stretched triangle.

The aspect ratio for a square or an equilateral triangle is 1, which is the ideal and lowest value.



Edge ratio (Practical)

For structured meshes or prism layers (with unstructured meshes), we often need very thin cells for boundary layers, that can lead to high aspect ratio cells (1,000 to 10,000).

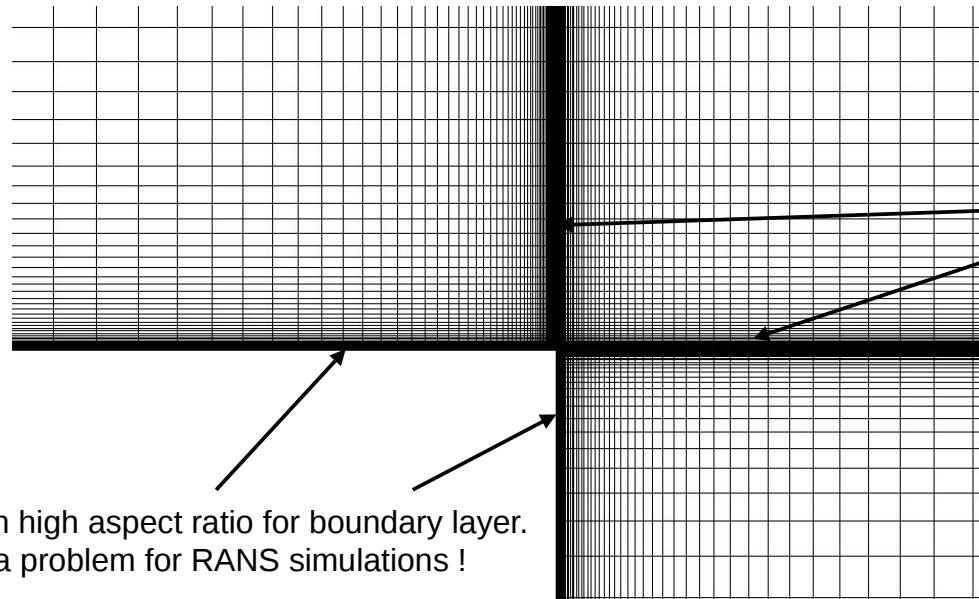
This is acceptable for RANS or URANS simulations, but not for LES or DNS simulations (due to additionnal quality criteria).

On the contrary, outside the boundary layers and especially in presence of gradients, aspect ratios should not exceed 20-50 to avoid :

- discontinuities in computed fields (velocity, pressure ...)
- difficulties to converge correctly (the computation is longer, because it need more iterations)
- sometimes none physical result for a few cells

But this last criterion is sometimes difficult to reach with structured mesh, because of the size of thin cells (ex : boundary layers) can propagate through the computation domain.

Example of high aspect ratio cells in a block structured mesh :



The size of thin cells propagates outside the boundary layer and generates cells with high aspect ratio.

Growth Rate (Theory)

Growth rate is another primary quality measure for a mesh.

It is the ratio of size between two neighbouring cells (biggest size over smallest size).

The name of this parameter depends on the software. But rules remain the same.

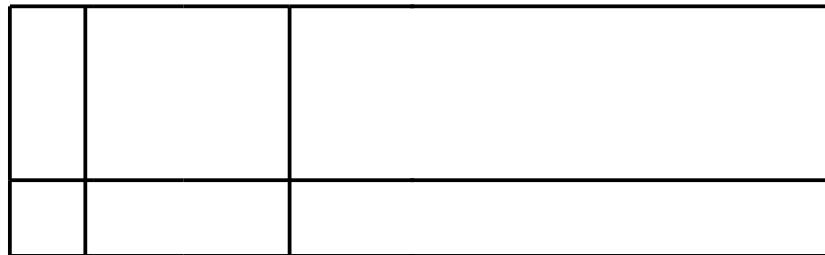
The default value in many meshing softwares is often great and must be decreased.

Maximum values for growth rates are :

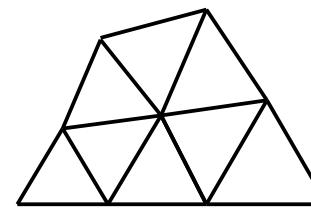
- 1.25 in boundary layers
- 1.3 outside the boundary layers



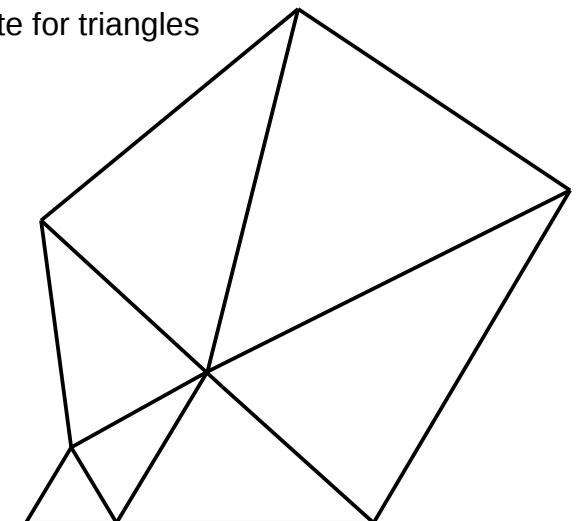
Good growth rate for quads



High growth rate for quads



Good growth rate for triangles



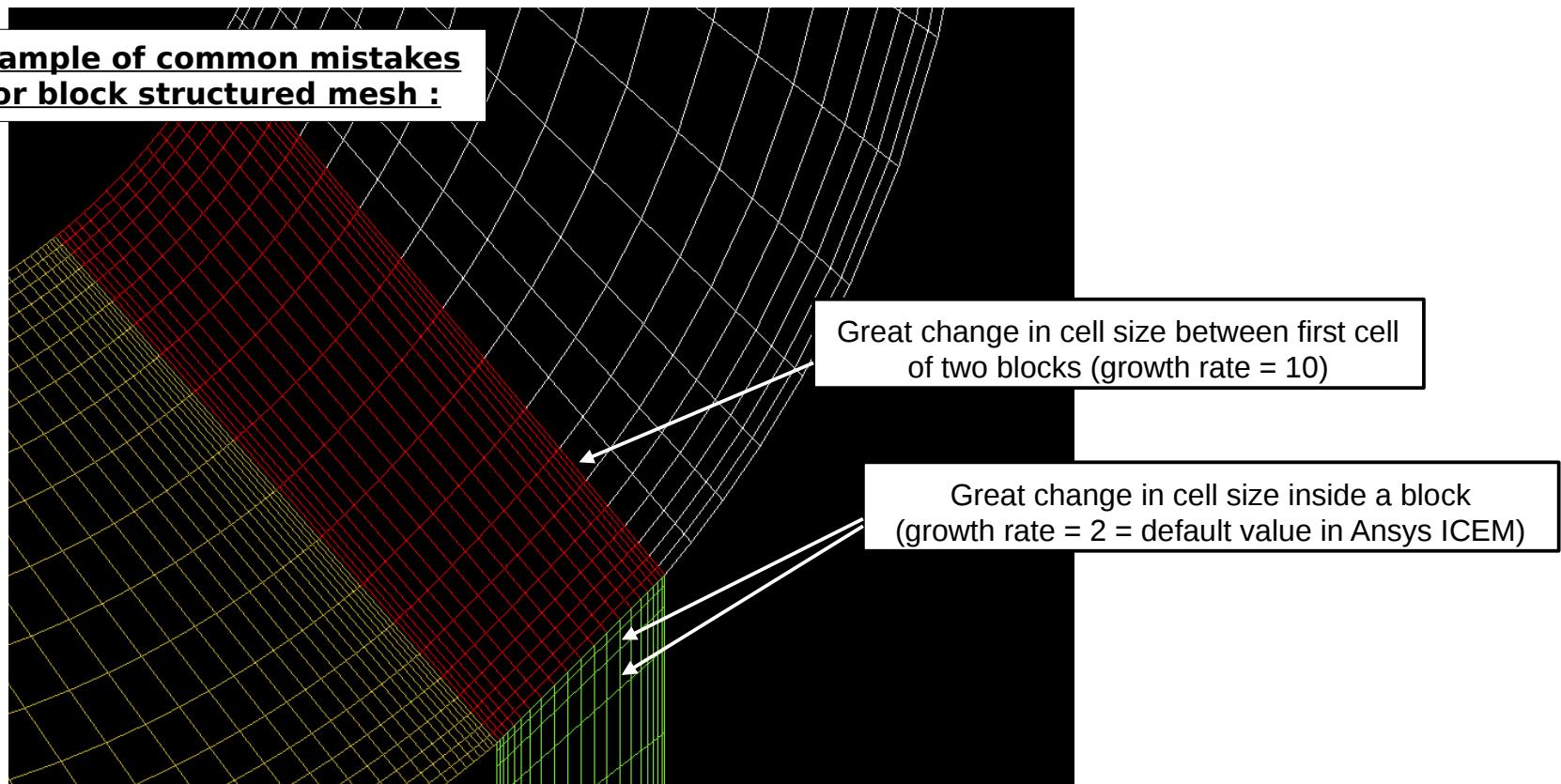
High growth rate for triangles

Growth Rate (Pratical 1/2)

For block structured meshes, some mistakes are often observed :

- junction between two blocks, a great change in the cell size for each first cell
- inside a block, the default value (often too high) is not changed

Example of common mistakes for block structured mesh :



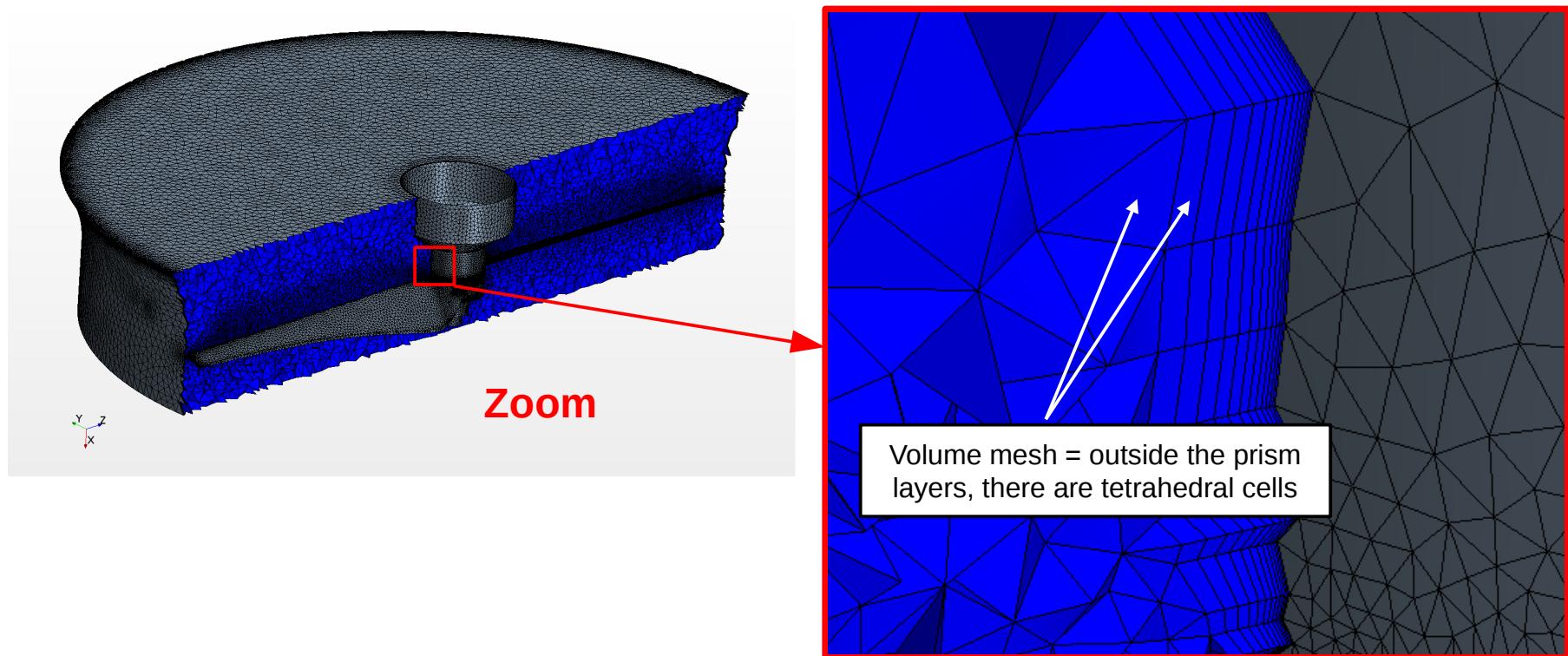
High growth rate values can lead to :

- errors for gradient calculations
- difficulties to converge correctly (the computation is longer because it need more iterations)

Growth Rate (Pratical 2/2)

For unstructured meshes, high growth rate are often observed between the prism layers and the « core » cells.

As a consequence the computation of gradients near the boundary layer can affect flow detachment.



It is difficult to avoid this kind of high aspect ratios in unstructured meshes.

Tools exist in some meshing softwares to solve this problem, but considerably increase the number of cells.

Meshing for Boundary Layers

Mesh for Boundary Layers : generalities

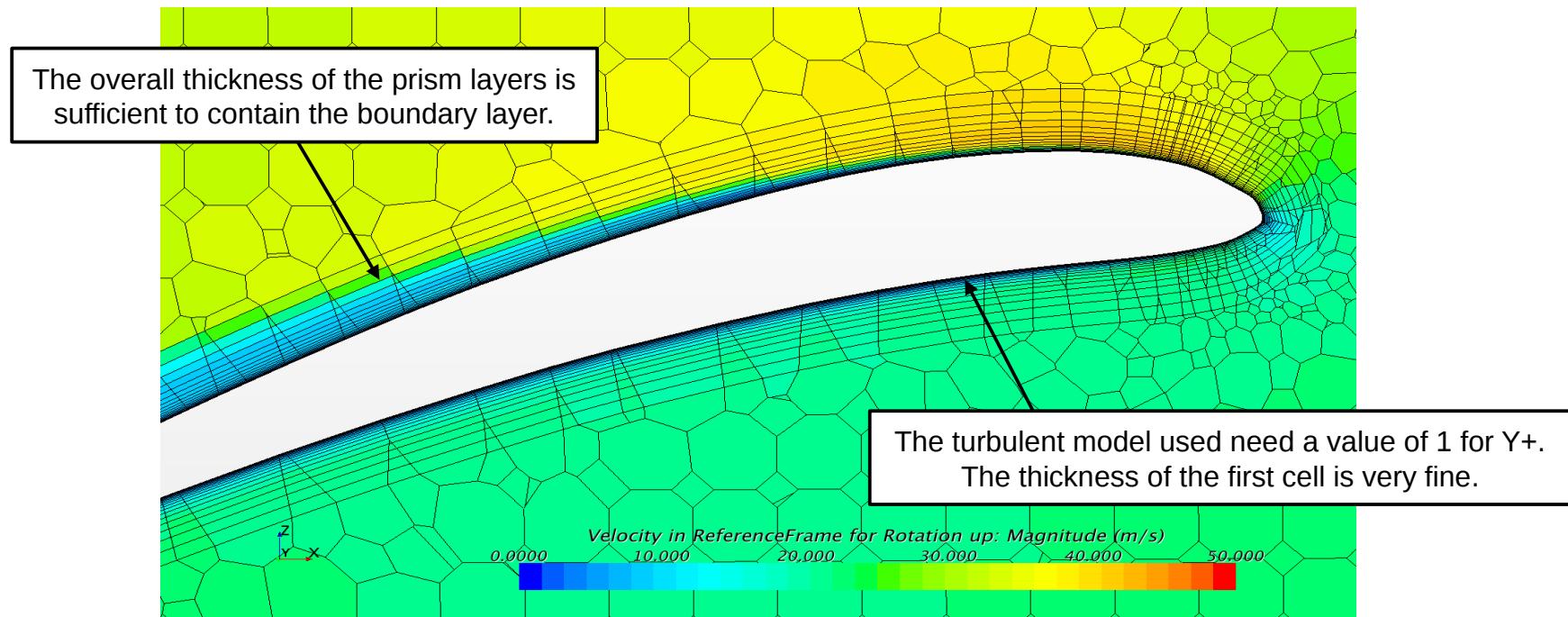
In RANS simulations, when the computation results strongly depends on the boundary layer behavior (ex : flow detachment, pressure drop mainly depending on the viscous effects ...), the mesh for the discretization of these boundary layers is very important.

In that case, some rules must be followed:

- the first cell in contact with the wall must be thin enough to obtain $Y+$ values in the range required by the near wall treatment ;
- in unstructured meshes, the prism layers must contain the boundary layer.

Before building the mesh for boundary layers, we need to calculate :

- the wall distance and then the thickness of the first cell (to aim a good value of $Y+$) ;
- in unstructured meshes, the thickness of the boundary layer and then the overall thickness for the prism layers.



Mesh for Boundary Layers : Wall Y+ (1/2)

RANS Turbulence models in CFD softwares use near wall treatments for boundary layer computation.

Depending on the CFD software, these near wall treatments are :

- chosen by the user
- automatically chosen by the software.

When they build the mesh, users must be sure that Y^+ values will be in the range of the chosen near wall treatment.

$$\text{The equation of } Y^+ \text{ is : } Y^+ = \frac{y u_\tau}{\nu}$$

Where u_τ is the friction velocity

The important parameter for mesh building is the wall distance y .

To find y value we need to know :

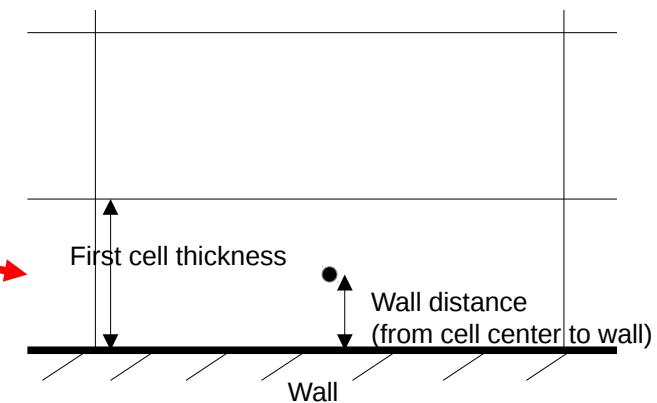
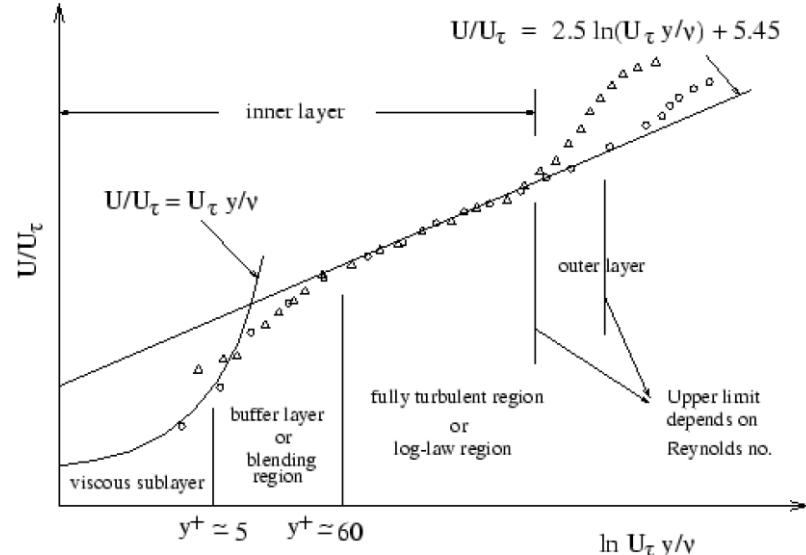
- free stream velocity
- the boundary layer length
- fluid properties (density and viscosity)

To estimate y , we can use the following tool :

<https://www.cfd-online.com/Tools/yplus.php>

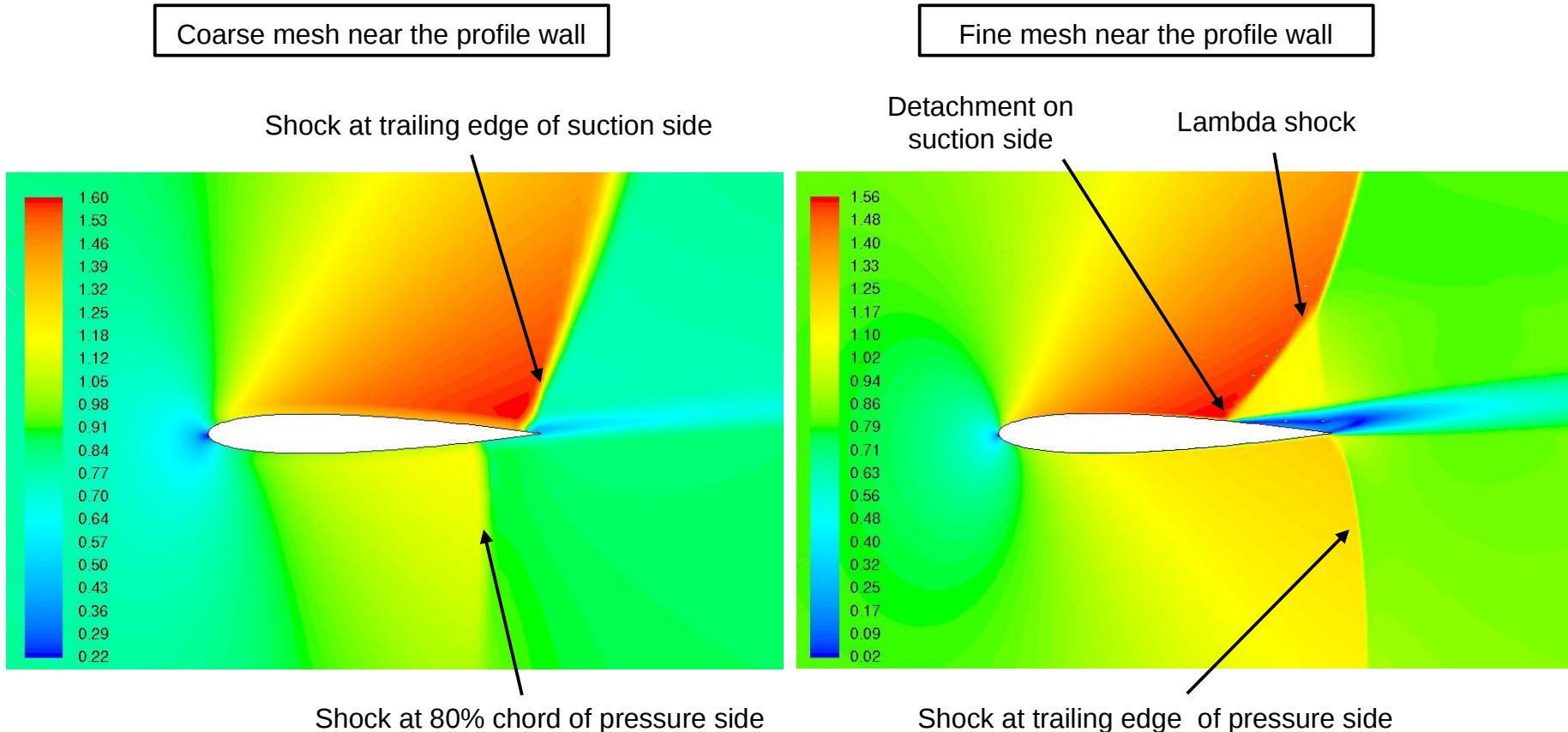
Finally, the first cell thickness (in contact with wall) is :

- **twice the wall distance y for volume finite solvers**
- the wall distance y for element finite solvers



Mesh for Boundary Layers : Wall Y+ (2/2)

Example of transonic flow around a NACA0012 profile with 5° incidence :



This example clearly shows the influence of the mesh size near the profile wall.

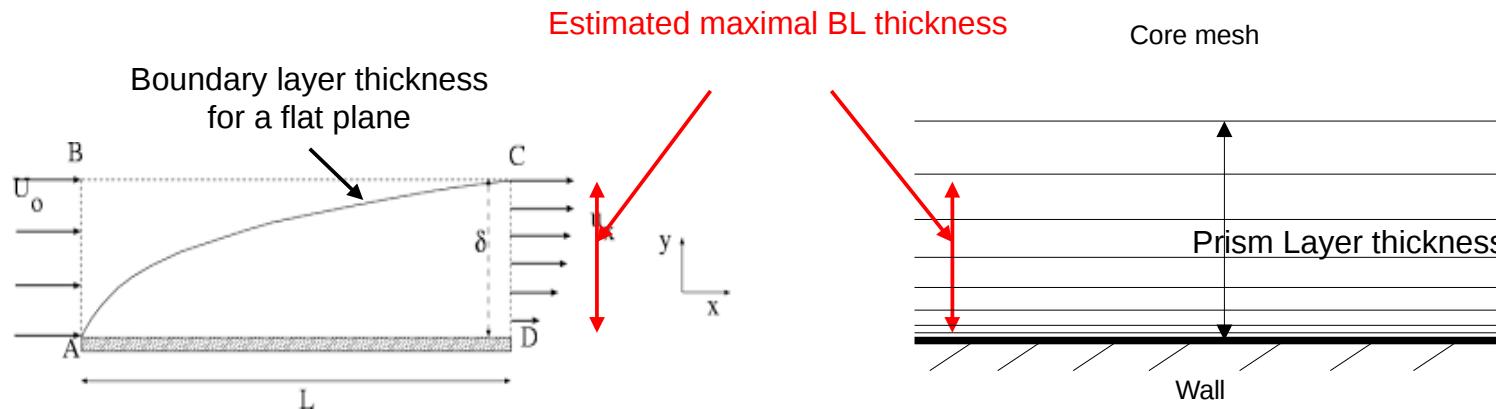
The flow detachment with the fine mesh changes considerably the shock nature and positions. The lift coefficient is divided by two between the two computations.

Mesh for Boundary Layers : BL Thickness

For unstructured meshes, it is important to evaluate the maximal boundary layer thickness.

The reason is we need to define the overall thickness for all prism layers which must contain the boundary layer.

However it is difficult to evaluate this maximal Boundary Layer thickness.
We can first use the equation for a flat plane and chose a higher value.



$$\text{For Laminar flow : } \delta \approx 5.0x / \sqrt{\text{Re}_x}$$

$$\text{With } \text{Re}_x = \rho u_0 x / \mu$$

$$\text{For turbulent flow : } \delta \approx 0.37x / \text{Re}_x^{1/5}$$

It is advised to check during the computation if the boundary layer is inside the prism layer.

Main Commercial Meshing Software

Commercial Meshing Softwares

Distributor	Mesher Software	Block Structured	Unstructured	Surface Wrapper
CD Adapco	Star CCM+		X	Powerful
Ansys	Gambit (no more distributed)	X	X	
	ICEM	X	X	Weak
	Mesher	X	X	Weak
Numeca	Fluent Meshing		X	Powerful
	IGG	X		
	Auto-Grid (for Turbomachinery)	X		
	Hexpress		X	Weak
	Hexpress Hybrid		X	Powerful

A surface wrapper is a tool to repair and close volumes.
A computation domain need to be closed to be meshed.

Questions ?