pyFluent | watertight网格生成

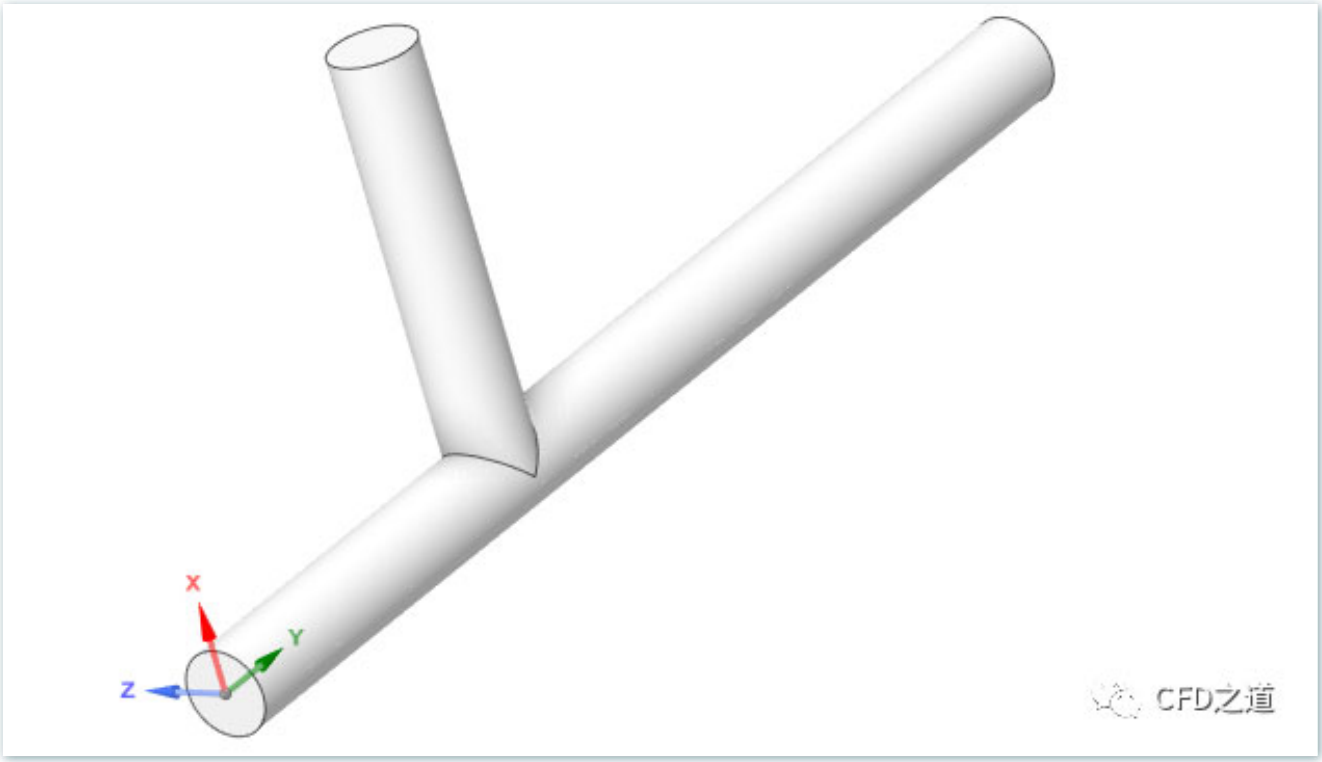
原创 流沙CAE CFD之道 2022-07-11 08:40 发表于四川

收录于合集
#pyFluent
4个

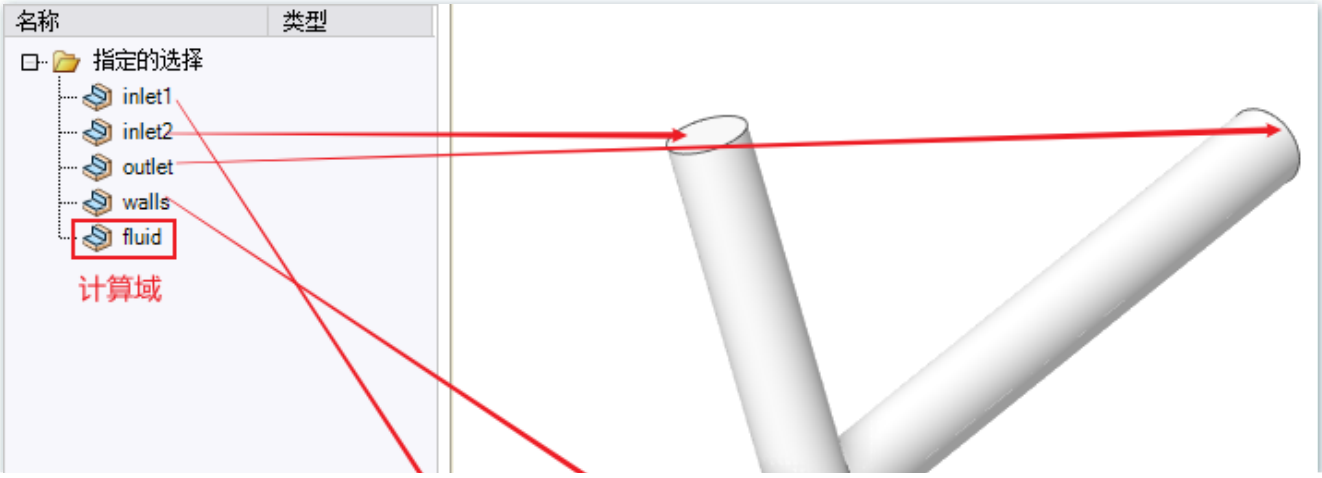
本文演示利用pyFluent导入几何模型，并采用watertight Geometry工作流生成计算网格的代码片段。

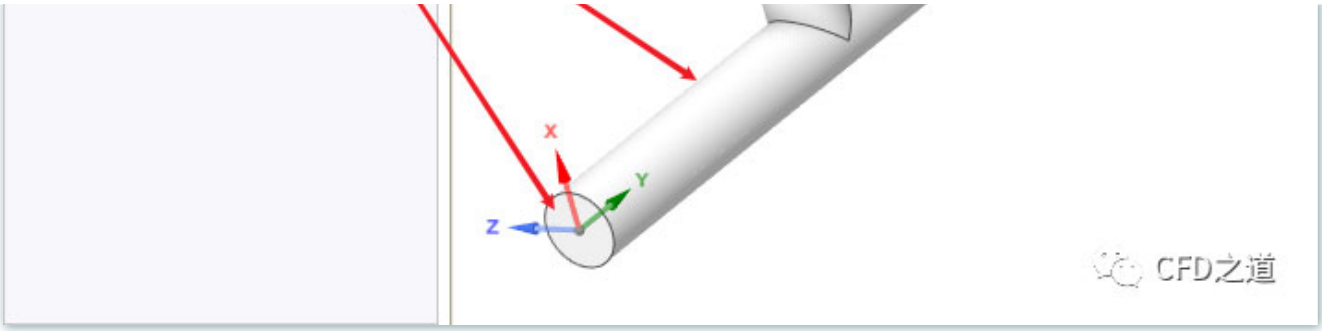
1 几何模型

几何模型在SpaceClaim中创建。



需要提前做好边界命名。若需要在后面的网格生成过程中指定特定区域网格尺寸、边界类型、边界层网格等，则需要在几何中进行边界命名，否则到后面没法识别边界。





几何命名为 `TeePipe.scdoc` 。

2 pyFluent代码

- 在几何文件相同的路径下创建文件mesh.ipynb，利用vs code打开

2.1 启动Fluent Meshing

- 输入下面的代码启动Fluent Meshing

```
import ansys.fluent.core as pyfluent
# 启用meshing模式，采用双精度，24个处理器
session = pyfluent.launch_fluent(meshing_mode=True,precision='double',processor_count = 24)
```

这里设置 `meshing_mode=True` ， 则表示以Meshing模式运行。

2.2 导入几何模型

输入下面的代码并运行。

```
geom_filename = 'TeePipe.scdoc'
session.meshing.workflow.InitializeWorkflow(WorkflowType='Watertight Geometry')
session.meshing.workflow.TaskObject['Import Geometry'].Arguments = dict(FileName = geom_filename,
session.meshing.workflow.TaskObject['Import Geometry'].Execute()
```

运行结果如下图所示，表示几何文件导入成功。

```
    attaching to assembly done in 2.276 [s]
    processing assembly 'D:\\TeePipe\\TeePipe.scdoc' with ID 1 and reference key '<?xml version="1.0"
encoding="utf-16"?>
    <IdTable xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <Table>
        <Entry>
```

```
...
analyzing boundary connectivity...done.

---------------- Import of TeePipe, consisting of one single part/object , complete.

True
```

CFD之道

2.3 设置局部尺寸

本案例涉及的几何尺寸分布比较均匀，不需要设置局部尺寸。采用下面的代码使用默认设置。

```
session.meshing.workflow.TaskObject['Add Local Sizing'].AddChildToTask()
session.meshing.workflow.TaskObject['Add Local Sizing'].Execute()
```

输出结果为 `True` 。

2.4 生成面网格

采用下面的代码设置面网格尺寸，并划分面网格。

```
session.meshing.workflow.TaskObject['Generate the Surface Mesh'].Arguments={'CFDSurfaceMeshContro
session.meshing.workflow.TaskObject['Generate the Surface Mesh'].Execute()
```

输出结果如下图所示。

```
20220704JV_23688/out165728164023688.tgf'
    converting file 'TeePipe.scdoc.pmdb' (1 of 1) from Workbench to FLTG using output path
'd:\\TeePipe\\FM_XTZJ-20220704JV_23688'
    importing data ...
    importing meshing model from PartMgr from file 'd:\\TeePipe\\FM_XTZJ-
20220704JV_23688\\\\TeePipe.scdoc.pmdb' ...
    setting enclosure and symmetry processing to False
    setting transfer named selections to True
    setting named selection prefixes to ''
...
---------------- After Surface mesh, the model consists of 1 fluid/solid regions and 0 voids.

---------------- Surface Meshing of TeePipe complete in  0.45 minutes, with a maximum skewness of
0.44.

True
```

CFD之道

2.5 几何描述

采用下面的代码进行几何描述。

```
session.meshing.workflow.TaskObject['Describe Geometry'].UpdateChildTasks(
    SetupTypeChanged=False
)

session.meshing.workflow.TaskObject['Describe Geometry'].Arguments=dict(
    SetupType='The geometry consists of only fluid regions with no voids'
)

session.meshing.workflow.TaskObject['Describe Geometry'].UpdateChildTasks(
    SetupTypeChanged=True
)

session.meshing.workflow.TaskObject['Describe Geometry'].Execute()
```

输出结果如下图所示。

```
---------------- Velocity-inlet zone type was automatically assigned to zones containing the string
'inlet'.

---------------- Pressure-outlet zone type was automatically assigned to zones containing the string
'outlet'.

True
```

CFD之道

2.6 更新边界

执行以下代码更新边界。

```
session.meshing.workflow.TaskObject['Update Boundaries'].Arguments={
    'BoundaryLabelList':['inlet1'],
    'BoundaryLabelTypeList':['velocity-inlet'],
    'BoundaryLabelList':['inlet2'],
    'BoundaryLabelTypeList':['velocity-inlet'],
    'BoudaryLabelList':['outlet'],
    'BoundaryLabelTypeList':['pressure-outlet']
}
session.meshing.workflow.TaskObject['Update Boundaries'].Execute()
```

运行结果如下所示。

```
Illegitimate input at /UpdateBoundaries; BoudaryLabelList
Illegitimate input at /UpdateBoundaries; BoudaryLabelList
Illegitimate input at /UpdateBoundaries; BoudaryLabelList

---------------- Boundary Conditions Updated

True
```

CFD之道

2.7 更新区域

本案例区域采用默认设置。

```
session.meshing.workflow.TaskObject['Update Regions'].Execute()
```

2.8 添加边界层

添加默认边界层参数。

```
session.meshing.workflow.TaskObject['Add Boundary Layers'].AddChildToTask()
session.meshing.workflow.TaskObject['Add Boundary Layers'].InsertCompoundChildTask()
session.meshing.workflow.TaskObject['smooth-transition_1'].Arguments={
    'BLControlName' : 'smooth-transition_1',
}
session.meshing.workflow.TaskObject['Add Boundary Layers'].Arguments={}
session.meshing.workflow.TaskObject['smooth-transition_1'].Execute()
```

2.9 生成体网格

利用下面的代码指定体网格尺寸并生成体网格。

```
session.meshing.workflow.TaskObject['Generate the Volume Mesh'].Arguments={
    'VolumFill': 'poly-hexcore',
    'volumeFillControls':{
        'HexMaxCellLength': 1.0,
    },
}

session.meshing.workflow.TaskObject['Generate the Volume Mesh'].Execute()
```

输出结果如下图所示。

```
processing scoped prisms...
    starting orientation...done.
    setting prism growth...done.
done.
Identifying Topology...

Generating Prisms...

Generating initial mesh...
.

...

---------------- The volume meshing of teepipe--- is complete.

Illegitimate input at /GenerateTheVolumeMeshWTM;

True
```

CFD之道

2.10 检查网格

输入下面的命令可以检查网格。

```
session.meshing.tui.mesh.check_mesh()
```

检查结果如下图所示。

```
Domain extents.
  x-coordinate: min = -1.000000e+01, max = 1.000000e+02.
  y-coordinate: min = 0.000000e+00, max = 2.900000e+02.
  z-coordinate: min = -1.000000e+01, max = 1.000000e+01.
Volume statistics.
  minimum volume: 8.087782e-04.
  maximum volume: 7.823082e-02.
    total volume: 1.198280e+05.
Face area statistics.
   minimum face area: 4.044900e-04.
   maximum face area: 1.916073e-01.
   average face area: 4.376239e-02.
Checking number of nodes per edge.
Checking number of nodes per face.
Checking number of nodes per cell.
Checking number of faces/neighbors per cell.
Checking cell faces/neighbors.
Checking isolated cells.
Checking face handedness.
Checking periodic face pairs.
Checking face children.
```

```
Checking face zone boundary conditions.
Checking for invalid node coordinates.
Checking poly cells.
...
Checking zones.
Checking neighborhood.
Checking modified centroid.
Checking non-positive or too small area.
```

CFD之道

2.11 保存网格

可以保存网格。

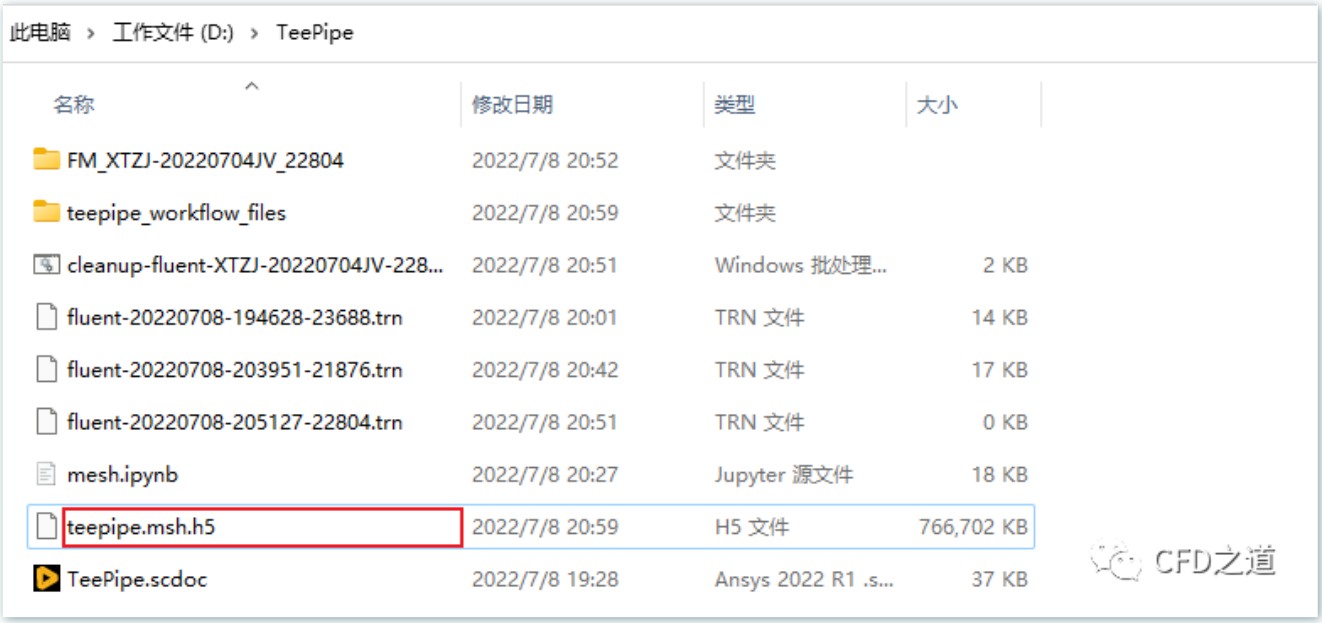
```
session.meshing.tui.file.write_mesh('teepipe.msh.h5')
```

输出结果如下图所示。

```
Done.Writing "teepipe.msh.h5" ...
writing 4 node zones
writing 4 edge zones
writing 5 face zones
writing 1 cell zones
done.Copying the required intermediate mesh files into teepipe_workflow_files
Done.
```

CFD之道

此时当前路径下有保存的网格文件。



2.12 结束

可以选择用下面的程序关闭Fluent：

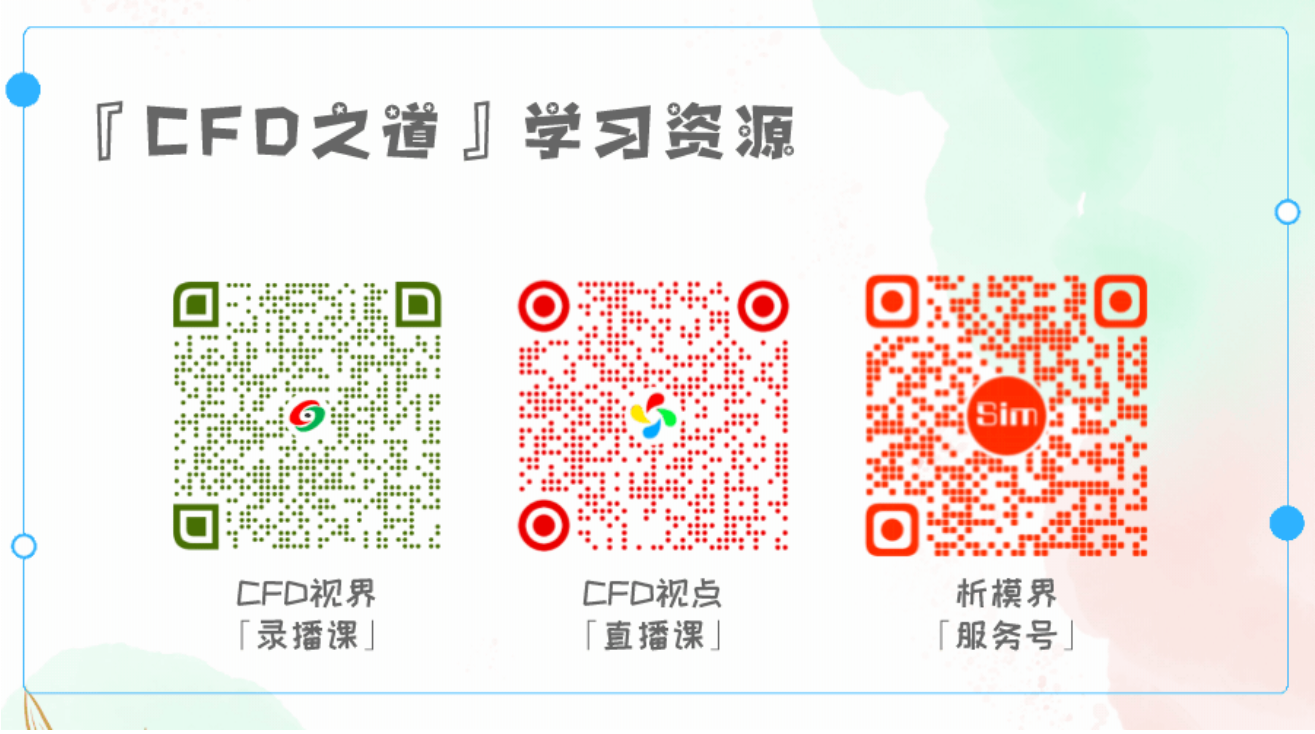
```
session.exit()
```

也可以使用下面的代码切换到solution模式，并执行后续的设置计算过程：

```
session.meshing.tui.switch_to_solution_mode("yes")
```

切记在事情干完后利用 `session.exit()` 退出Fluent。

（本文结束）



收录于合集 #pyFluent 4

上一篇 · pyFluent | 设置Fluent计算参数

喜欢此内容的人还喜欢

B端交互 | 从组件、栅格和响应式入手来制定全局框架

超人的电话亭

Python 代码转 Latex 公式，这个开源库用一行代码帮你搞定

小白学视觉

薅一个免费的在线Rstudio

基因学苑