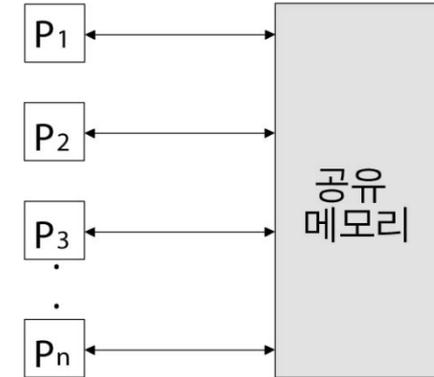


CFD-ACE+, Utility를 사용한 병렬해석 설정

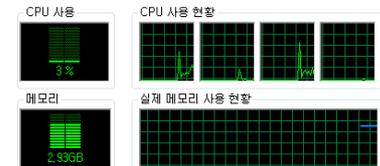
(주) 경원이앤씨

• 병렬 프로그래밍의 개요

- 한 개의 계산을 여러 프로세스에서 나누어서 실행하는 기능으로서 한 개의 프로세스에서 계산하는 것보다 **계산시간을 단축 가능**
- 프로세스간에 주고 받을 정보를 MPI 기능을 이용하여 주고 받으면서 **동시 계산 진행** → **데이터 전송 시간이 있기 때문에 프로세스 개수와 계산 시간이 정확히 반비례하지는 않는다**
- 현재 하드웨어의 발전으로 워크스테이션이나 개인용 PC에서도 병렬 계산이 가능한 멀티코어 프로세스를 탑재하고 있음

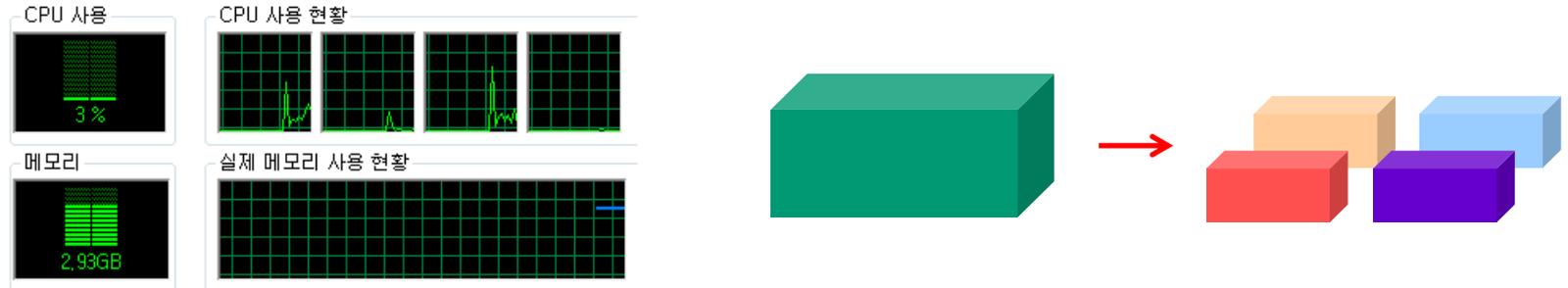


수퍼컴퓨터



멀티코어 Workstation

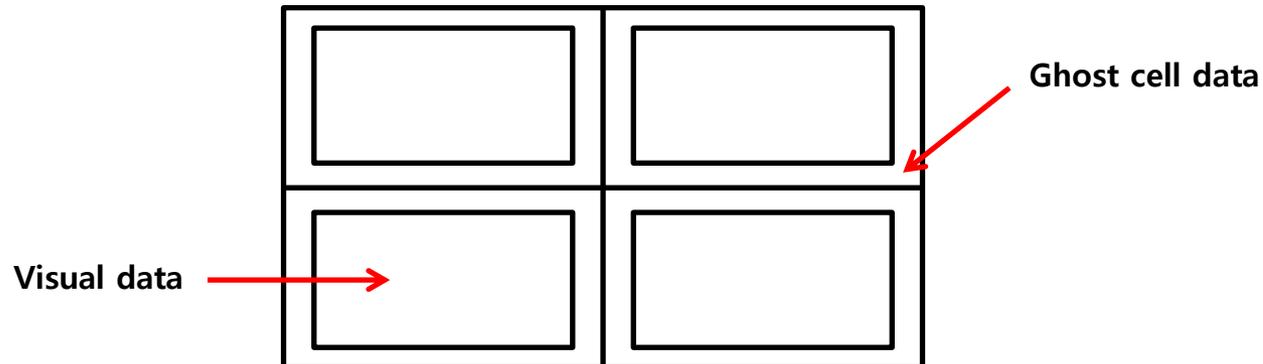
- 계산 영역 분할을 통한 병렬화 필요

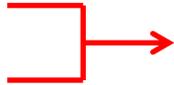


- 분할 영역의 개수와 해석에 사용할 컴퓨터 프로세서의 개수를 일치시켜야 함

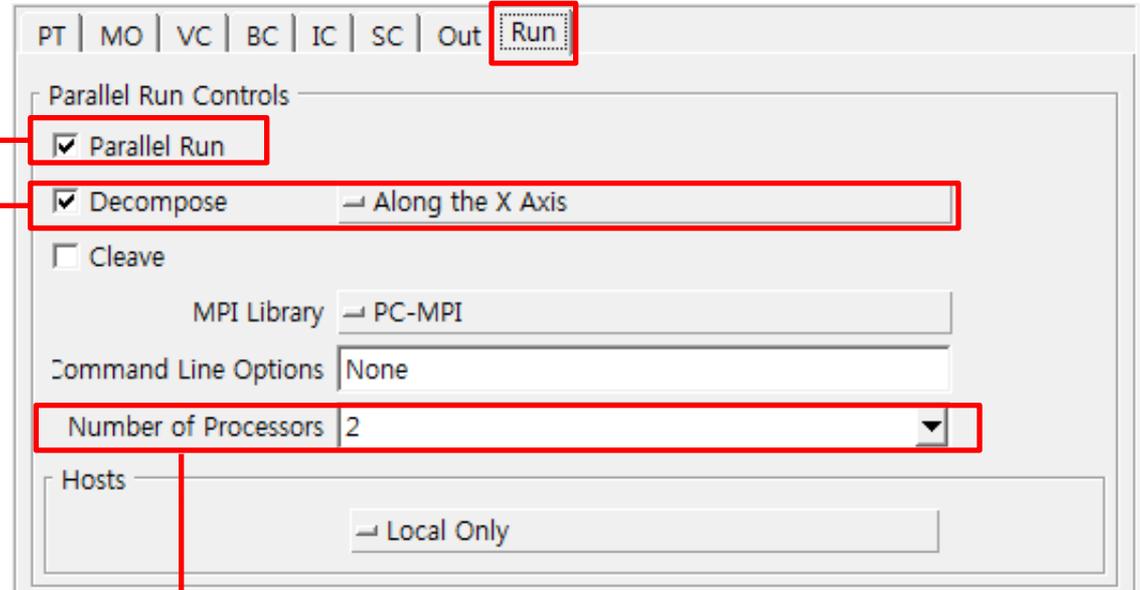
- 가장 격자(Ghost cell) 개념

- 이웃하는 분할영역의 경계 정보를 제공하기 위하여 각 분할영역의 경계를 따라 가상 격자(Ghost cell)가 추가됨



- Limitation
 - Arbitrary interface boundary conditions
 - Cyclic boundaries separated by a zonal interface
 - Stress module
 - Two-phase module
 - Thin-wall boundary conditions
 - Total heat source
 - Fixed Mass flow inlet 
- 면이나 체적 기반으로 값이 지정되기 때문에 가급적이면 이러한 조건이 설정된 면이 분할되지 않도록 해야 함
- 병렬 계산의 적용
 - 기존에는 별도로 구축된 병렬 머신에서 작업이 이루어졌으며, linux의 ssh를 통한 script 명령어 형식이 사용되었음 → 일반 사용자가 접근이 어려움
 - 현재 CFD-ACE-GUI 상에서 간단한 클릭만으로 병렬 계산이 적용 가능

- 병렬 계산 설정



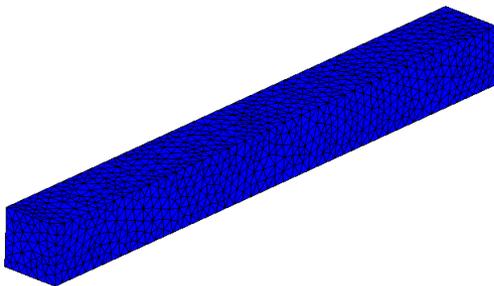
병렬 계산 기능 선택

프로세서의 수에 맞게 도메인을 분할
(분할되어 있는 모델은 클릭 해제)

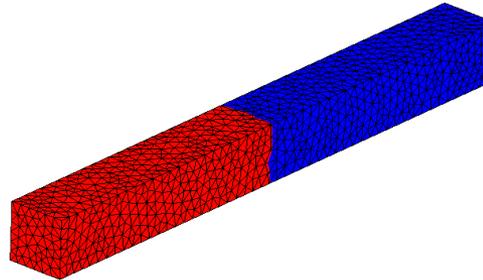
- Multilevel Graph Partition
- Along the X Axis
- Along the Y Axis
- Along the Z Axis
- Along the Cell Groups
- Along the Existing Topology

도메인 분할 방법

병렬 계산에 사용될
프로세서 개수 지정



Sim #1



Sim #2

- 계산이 시작되면서, 프로세서의 개수에 맞게 도메인이 분할 된다.

- DTF_decompose : 병렬 계산 시 해석 영역을 나눠 주는 방식 선택
 - CFD-ACE-GUI에서는 기본적인 옵션만 선택이 가능하며 utility를 사용하면 기본 옵션 이외에 여러 가지 옵션 선택 가능

```
관리자: C:\Windows\system32\cmd.exe - dtf_decompose
C:\Users\Wuser>dtf_decompose
Usage entered is incorrect.
You entered:
    dtf_decompose

Correct usage is:

dtf_decompose [-version] [-metis ! -cell_groups ! -orig_topo ! -x ! -y ! -z ! -w w1 w2...] [-file_out outFile.DTF] [-even] [-combined] [-keepFF] [-inplace] [-w w1 w2...] [-file_out outFile.DTF] [-restart] inFile.DTF sim# num_procs

where:

  -version          prints version number

  -metis            decompose by METIS method <DEFAULT>
  -cell_groups      decompose by cell groups <ignores num_procs>
  -orig_topo        decompose along existing zones <ignores num_procs>
  -x                decompose along the X axis
  -y                decompose along the Y axis
  -z                decompose along the Z axis
  -wavefront        decompose using wavefront method

  -w w1 w2...       give weight factor for each process.
                    the total number of weight factors should be equal to num_procs
  -even             cause -x, -y, -z decomposition to be spatial balanced
                    instead of workload balanced
  -combined         combine volumes and boundaries together with identical
                    condition in the same decomposed zone
  -kp               keep process interface not combined if process interface
                    lies on the boundary face
  -keepFF           keep fluid-fluid interface inside of a zone

  -file_out outFile.DTF alternate DTF file to place decomposed simulation
                    if not specified, decomposition will go into inFile.DTF

  -inplace          do not use a temporary DTF file
                    <replaces structured zones immediately>

  -restart          force mapping of face data

  [-dmp!-nodmp]    enable/disable DMP parallel processing <default=disable>
  [-smp!-nosmp]    enable/disable SMP parallel processing <default=disable>

  inFile.DTF       DTF file to be decomposed
                    decomposition will be placed here unless -file_out is used
  sim#             simulation number to decompose
  num_procs        decompose for num_proc processes <zones>
                    not needed for -cell_groups and -orig_topo options

Press ENTER to exit script_
```

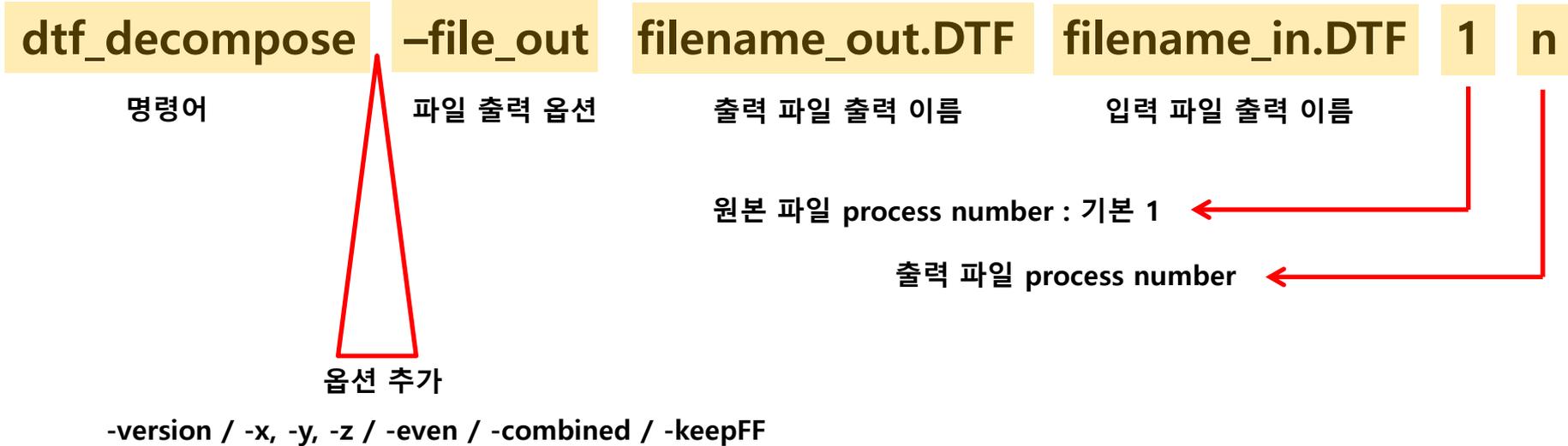
Windows command 창

dtf_decompose 입력 : decompose에 대한 각종 옵션을
확일 할 수 있음.

많이 사용되는 옵션

- ✓ -version : 사용할 버전 선택(-runver 2013.0)
- ✓ -x, -y, -z : 해석 영역을 분할 할 때 방향성을 갖게 함
- ✓ -even : -x, -y, -z의 방향에 따른 균일한 길이로 분할
- ✓ -combined : 같은 이름이 volume conditions, boundary conditions 를 묶어 줌 / decompose 명령으로 인한 분할된 영역을 다시 합쳐 줄 경우 사용 됨.
- ✓ -keepFF : fluid-fluid사이의 interface를 남겨 줌.
- ✓ -file out : 출력할 파일이름 지정

- DTF_decompose 사용 방법
 - 기본 옵션



- 실행 예

- Dtf_decompose -file_out HE_pipe_flow_N4.DTF HE_pipe_only.DTF 1 4

```
관리자: C:\Windows\system32\cmd.exe
Mapping the original file to the new file...
Working on processor 0
Working on processor 1
Postprocessing...
Done.
0

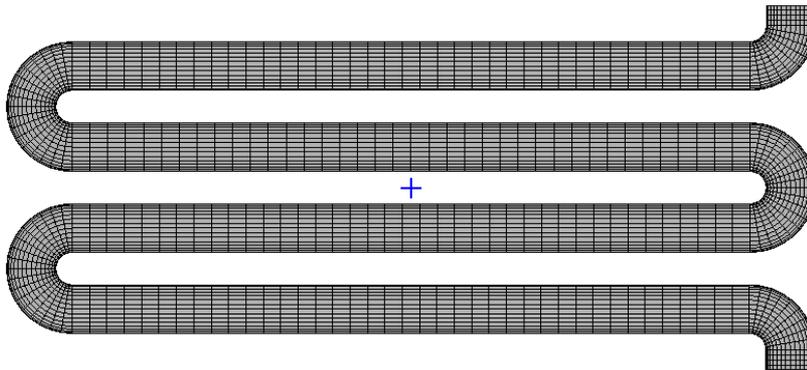
H:\WCFD-ACE_module\W2013_release\decompose\default>dtf_decompose -file_out HE_pipe_flow_N4.DTF HE_pipe_only.DTF 1 4
dtf_decompose U3.5.10 DTF library U7.8.6
Checking file existence...
Checking available disk space...
Running preliminary checks...
Preprocessing...
Creating partitions...
Mapping the original file to the new file...
Working on processor 0
Working on processor 1
Working on processor 2
Working on processor 3
Postprocessing...
Done.
0
H:\WCFD-ACE_module\W2013_release\decompose\default>
```

원본 파일 : HE_pipe_only.DTF

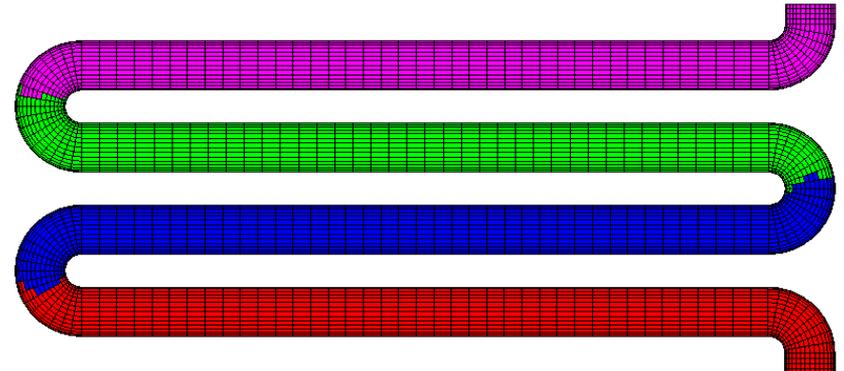


출력 파일 : HE_pipe_flow_N4.DTF

- Process number 4개로 분할한 경우

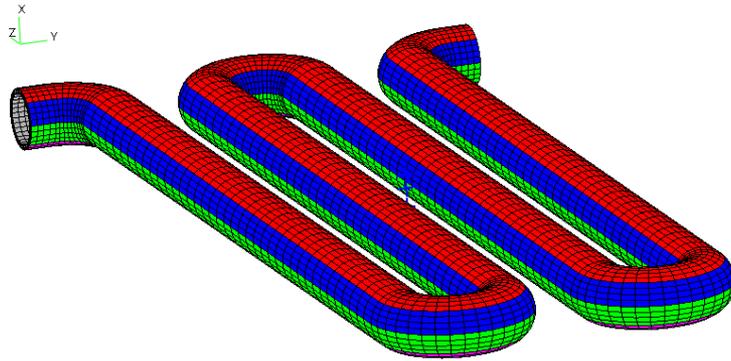


원본 파일 : HE_pipe_only.DTF

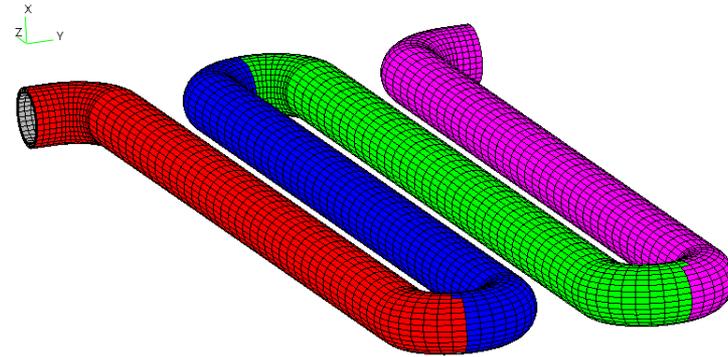


출력 파일 : HE_pipe_flow_N4.DTF

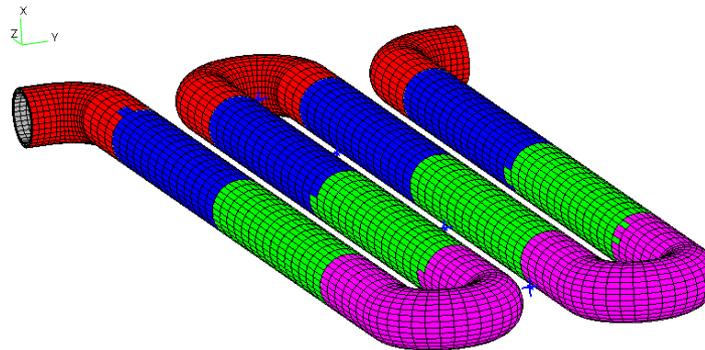
- 옵션 1(-x, -y, -z)
 - 분할 영역의 방향성 : -x, -y, -z
 - Dtf_decompse **-x** -file_out HE_pipe_flow_N4.DTF HE_pipe_only.DTF 1 4



-X 방향 분할

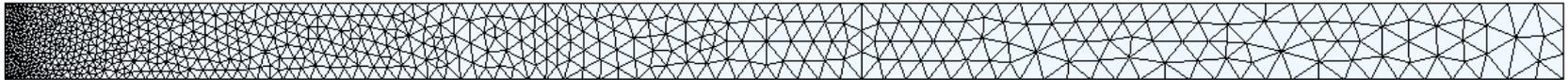


-Y 방향 분할



-Z 방향 분할

- 옵션 2 (-even)
 - $-x$, $-y$, $-z$ 의 방향에 따른 균일한 길이로 분할
 - 기본 옵션 - x 축 좌측방향으로 격자 밀집



x축 →

기본 격자 구조

- 기본 옵션 : $-x$ 추가 (Dtf_decompose $-x$ -file_out duct_n6.DTF duct.DTF 1 6)



Process #6 $-x$ 방향

- 기본 옵션 : $-x$ 추가, $-even$ 추가
- (Dtf_decompose $-x$ $-enve$ -file_out duct_n6.DTF duct.DTF 1 6)



Process #6 $-x$ 방향, $-even$

- 옵션 3(-combined)

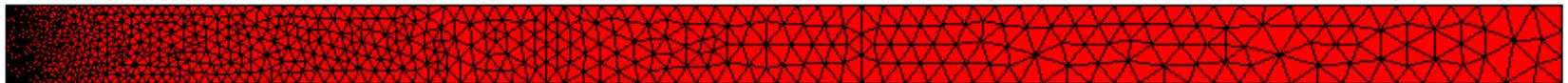
- Decompose 명령어를 이용하여 분할된 영역을 다시 1 process로 합칠 경우
- 명령어

- `dtf_decompose -combined -file_out duck_combined.DTF duct_n6.DTF 1 1`

Process #1로 설정 ←



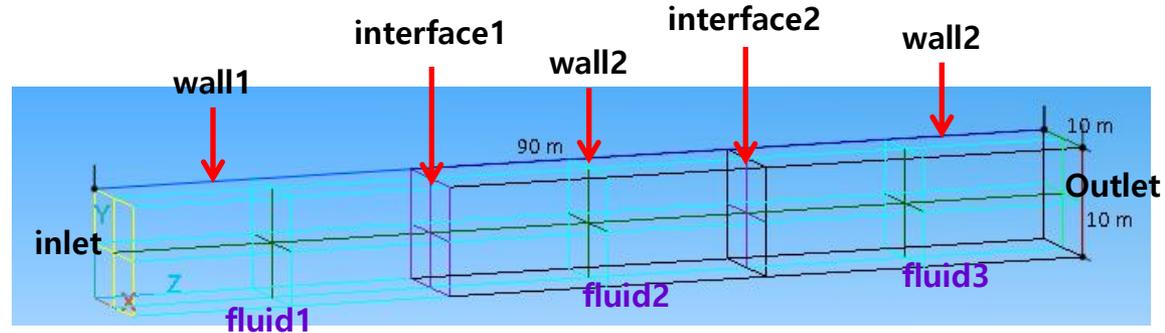
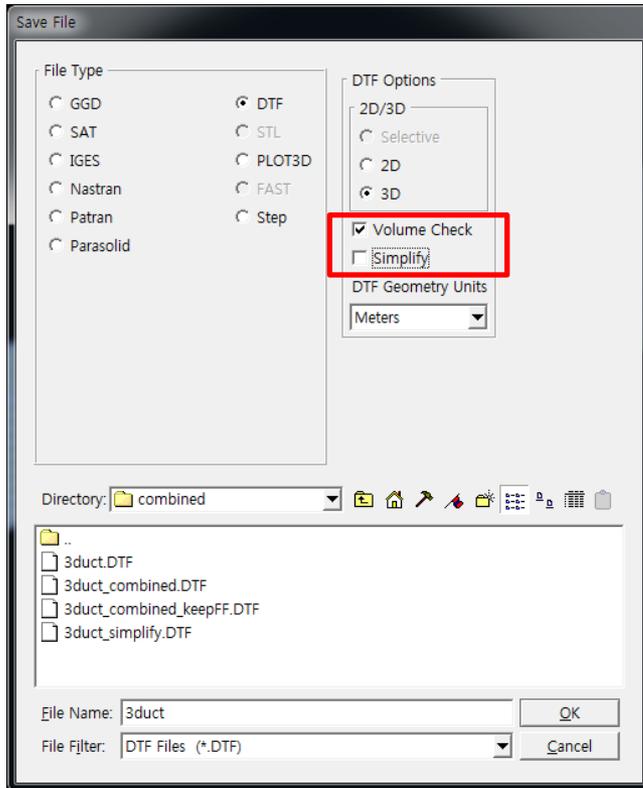
Process #6 -x 방향



Process #1

- Boundary/Volume conditions를 합칠 경우

- CFD-GEOM : default 설정으로 DTF를 저장 하였을 경우



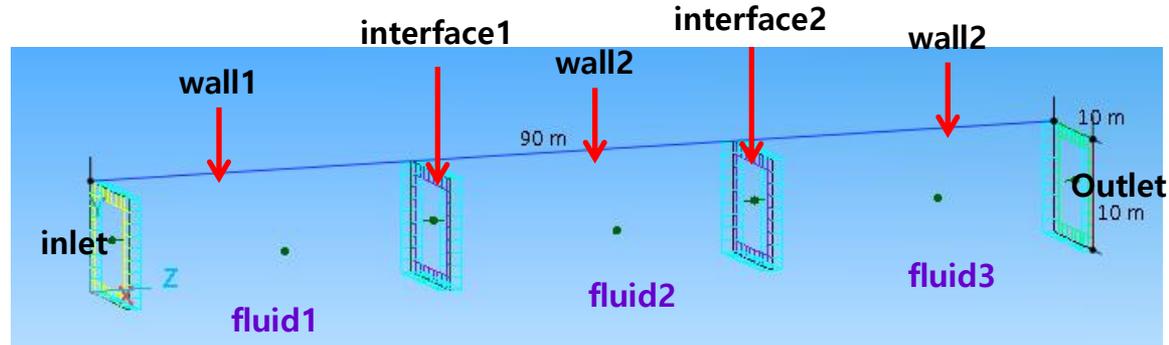
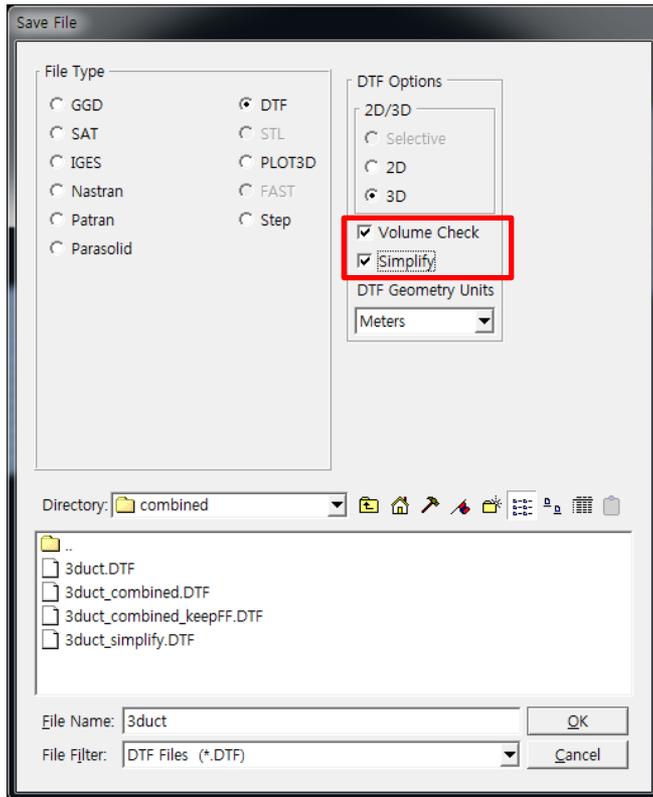
Default : CFD-GEOM에서 설정한 값 그대로 나타남

Boundary Name	BC Type	BC SubType	Blanked	General	Zone	Key
inlet	Inlet				1	13
interface1	Interface				1<=>2	164
wall1	Wall				1	152
wall1	Wall				1	154
wall1	Wall				1	155
wall1	Wall				1	153
interface2	Interface				2<=>3	165
wall2	Wall				2	156
wall2	Wall				2	158
wall2	Wall				2	159
wall2	Wall				2	157
outlet	Outlet				3	166
wall2	Wall				3	160
wall2	Wall				3	162
wall2	Wall				3	163
wall2	Wall				3	161

총 : 16개

Default : CFD-ACE-GUI의 boundary conditions

- CFD-GEOM simplify를 선택 하였을 경우



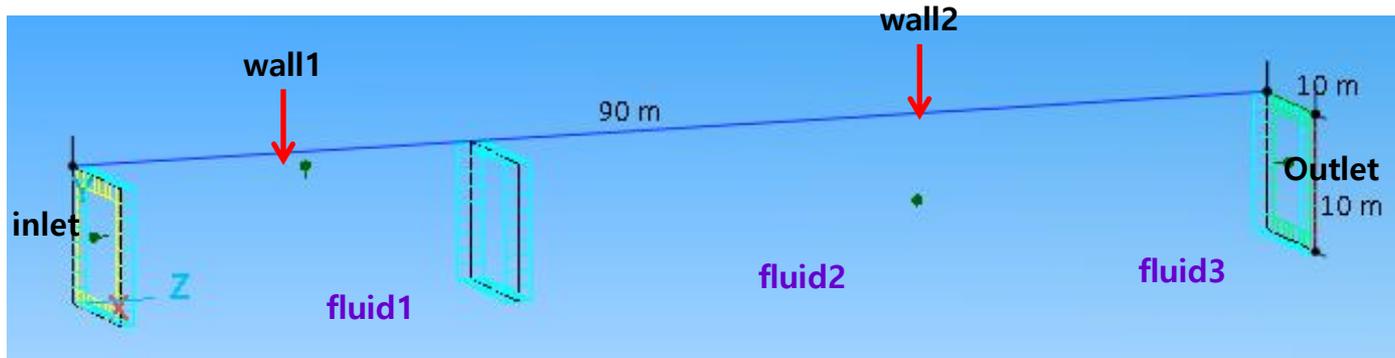
	Boundary Name	BC Type	BC SubType	Blanked	General	Zone	Key
	inlet	Inlet				1	4
	interface1	Interface				1<=>2	5
	wall1	Wall				1	6
	interface2	Interface				2<=>3	7
	wall2	Wall				2	8
	outlet	Outlet				3	9
	wall2	Wall				3	10

총 : 7개

- ✓ 인접한 wall1과 wall2는 1개의 wall로 합쳐 짐.
- ✓ Wall2의 좌측과 우측은 다른 이름의 volume condition으로 인하여 분리 됨
- ✓ simplify utility를 이용하여 Default로 설정된 파일을 simplify 옵션과 같이 이용 가능
- 예 : dtf_simplify in_file sim# out_file

- Decompose 명령어를 이용한 combined
- 명령어 :

- dtf_decompose **-combined** -file_out 3duck_combined.DTF 3duck.DTF 1 1



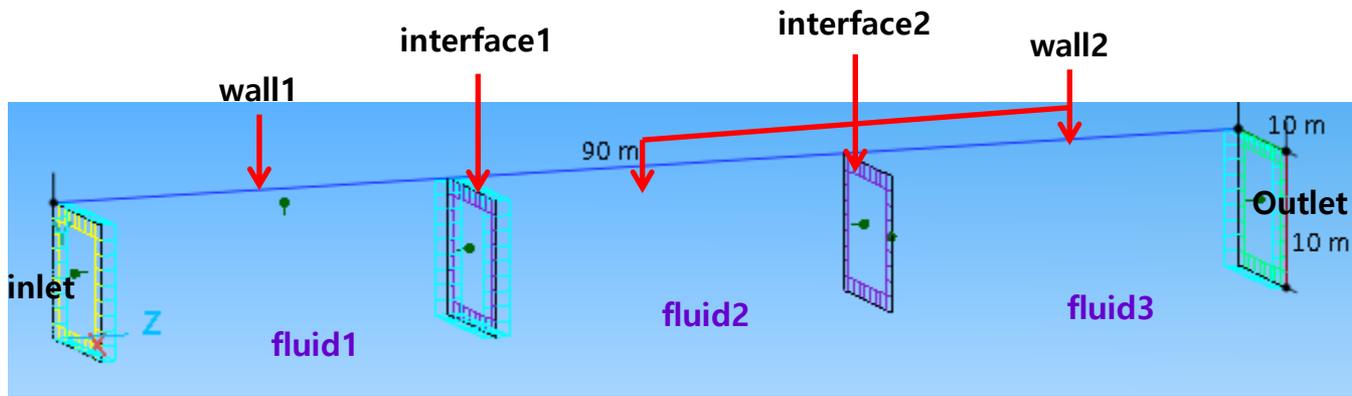
	Boundary Name	BC Type	BC SubType	Blanked	◆ General	Zone	Key
	outlet	Outlet	} 총 : 4개			1	1
	inlet	Inlet				1	2
	wall2	Wall				1	3
	wall1	Wall				1	4

- ✓ 같은 이름으로 지정한 이름은 전부 1개 boundary condition으로 합쳐 짐.
- ✓ Fluid-fluid 사이의 interface는 보존되지 않고 없어짐. (fluid-solid interface는 남아 있음)

- Decompose 명령어에서 combined과 keepFF를 함께 사용한 경우
- 명령어 :

• dtf_decompose **-combined -keepFF** -file_out 3duck_combined.DTF

3duck.DTF 1 1

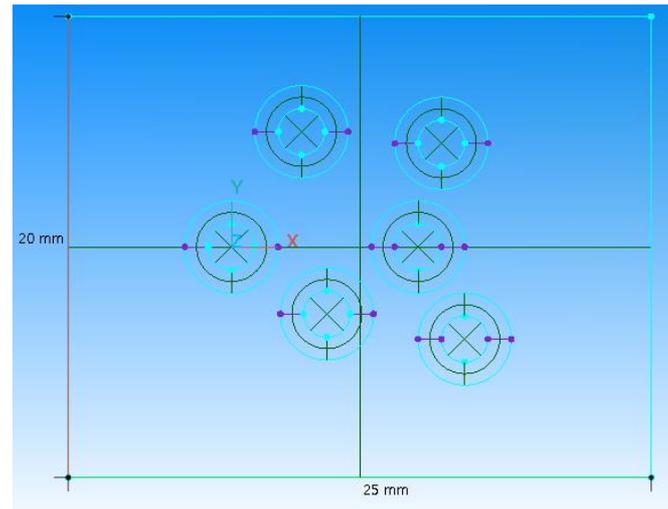
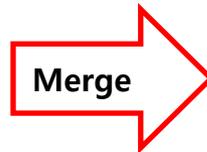
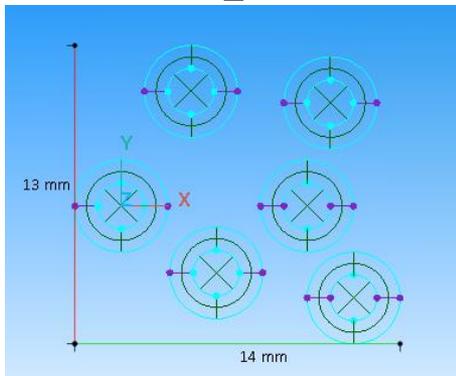
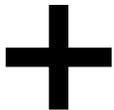
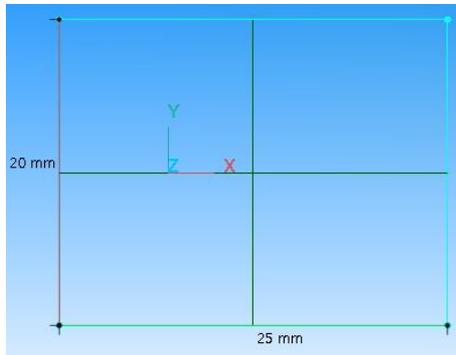


	Boundary Name	BC Type	BC SubType	Blanked	◆ General	Zone	Key
	outlet	Outlet				1	1
	inlet	Inlet				1	2
	wall2	Wall				1	3
	wall1	Wall				1	4
	interface2	Interface				1	5
	interface1	Interface				1	6

총 : 6개

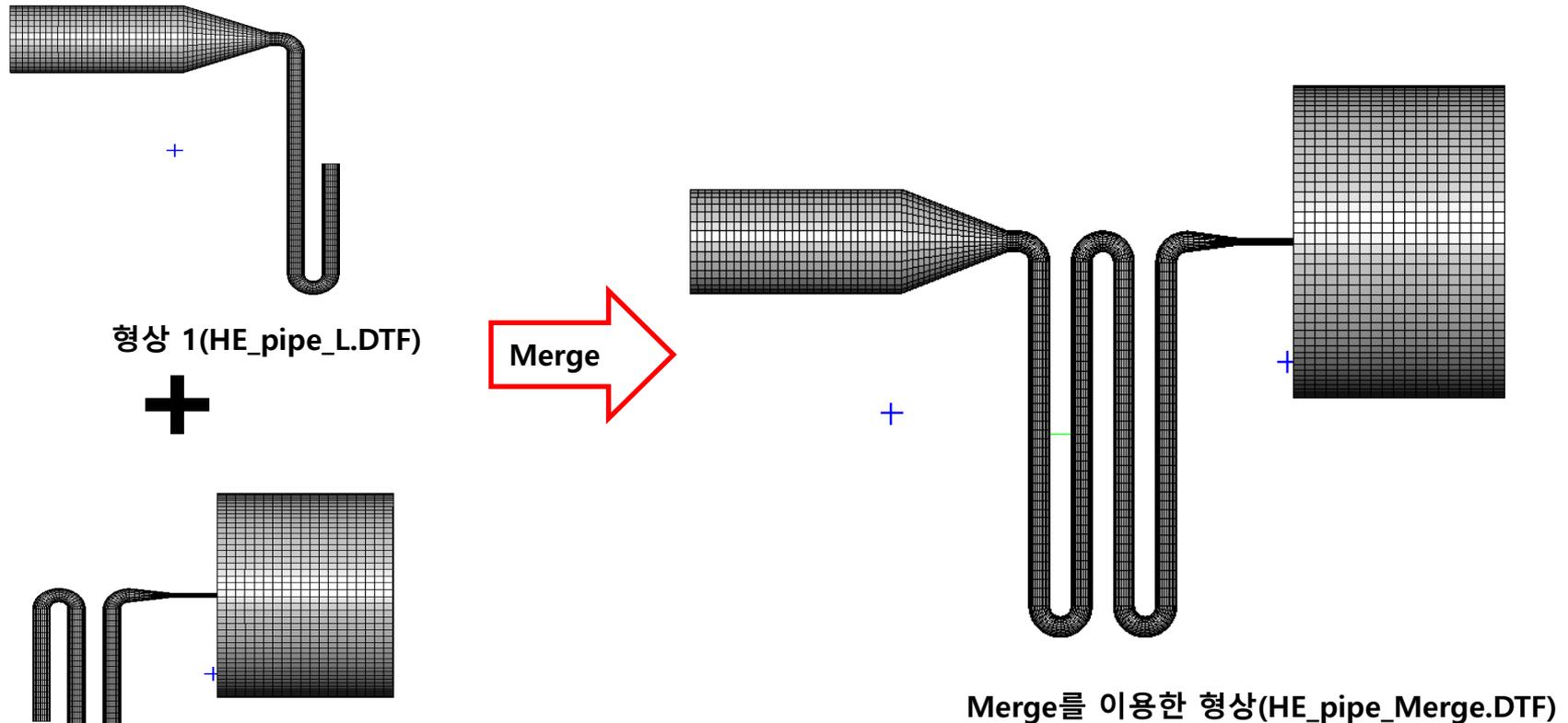
- ✓ 이름을 지정한 interface는 보존 됨
- ✓ 같은 이름의 wall2는 1개의 boundary condition으로 합쳐 짐

- 명령어 : dtf_merge
 - 2개 이상의 DTF파일을 합칠 경우에 사용 됨
 - 주로 chimera grid를 만들 경우 이용 됨.
 - 계산량이 많은 형상에 대하여 각각의 형상에 대하여 DTF를 만든 후 arbitrary interface로 합 칠 경우에 사용 됨.



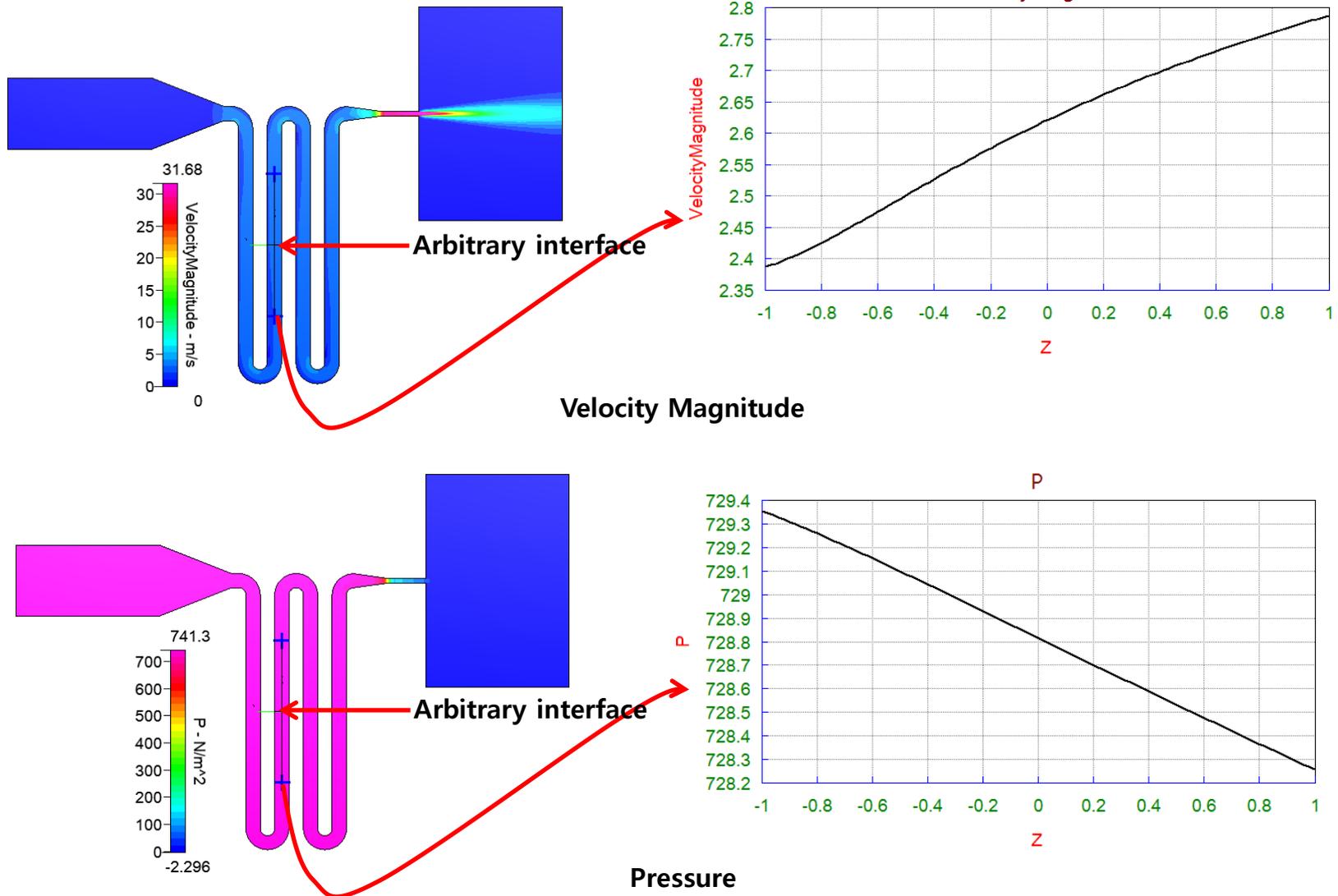
Merge를 이용한 chimera grid 생성

- 복잡한 형상에 대한 DTF merge
 - 형상1과 형상2의 좌표는 두 형상이 merge 된 경우와 일치해야 함
 - Merge된 형상이 만나는 부분은 arbitrary interface로 처리해야 함



예) Dtf_merge He_pipe_L.DTF HE_pipe_R.DTF HE_pipe_Merge.DTF
형상1, 형상2에 대한 DTF 새로 생성된 DTF

- 복잡한 형상에 대한 DTF merge 해석 결과



Thank you for your attention

KWEnC CAE solution provider

Mail to : support@kwenc.kr

Phone : +82-31-737-2987

Fax : +82-31-737-2989

Address : 704, Woolim Lions Valley, 311-3 Sangdaewon, Jungwon, Seongnam, Gyeonggi,
462-806 Korea

Copyright(c)KWENC, All right reserved.
