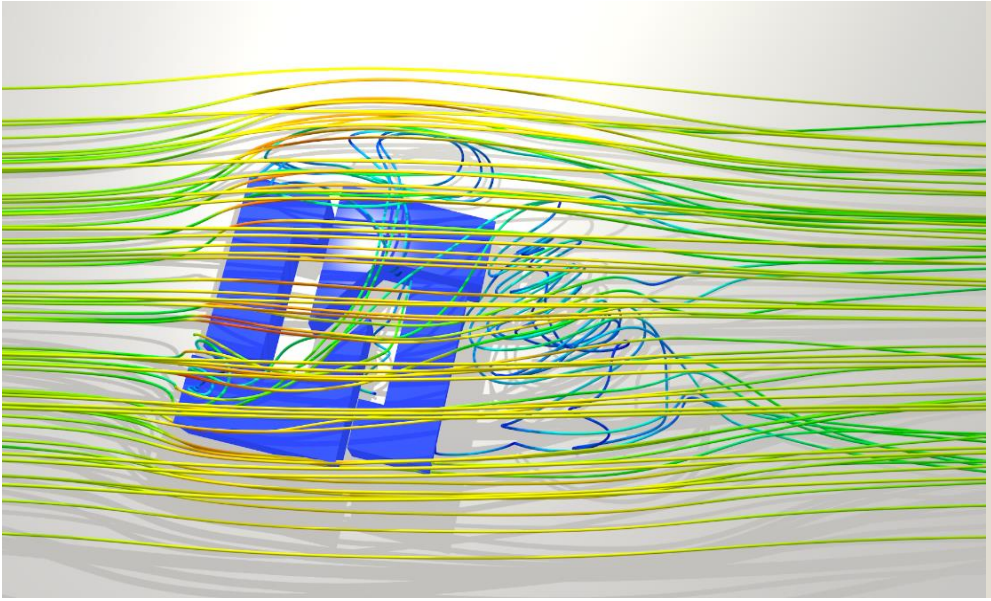
2021 한국전산유체공학회 추계학술대회, 서울대 시흥캠퍼스

BARAM-v6.3 upgrade



2021. 11. 11

㈜넥스트폼 김 병 윤

목차

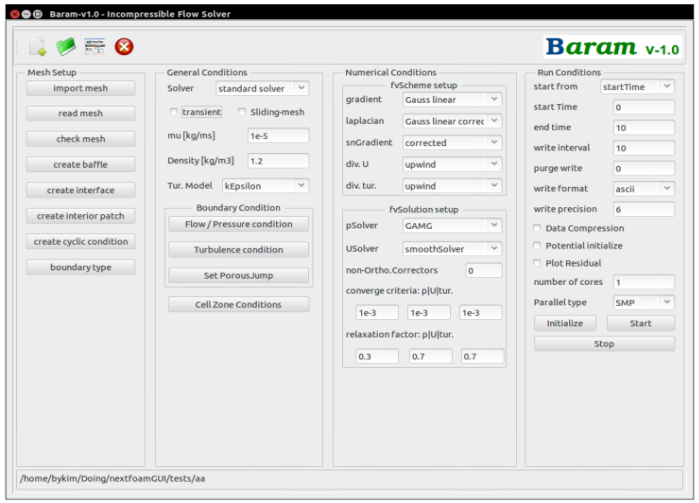
BARAM이란?

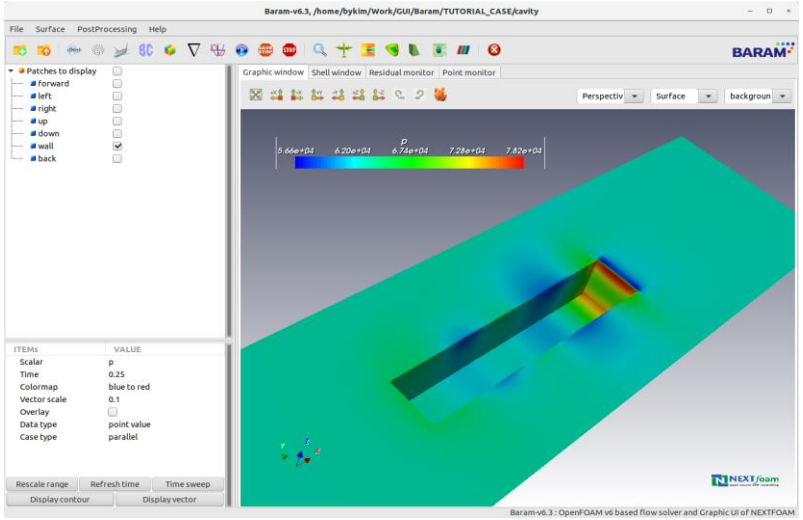
- 넥스트폼이 개발한 OpenFOAM® 기반의 CFD 패키지
- 공개소스 프로그램 GNU GPL

OpenFAOM	• The openFOAM Foundation
nextFoam6 TSLAeroFoam	• 넥스트폼의 OpenFOAM 패키지 • Solver, Library, API
GUI	• Pre / post

History

2015 v1.0 → 2015, v2.0 → 2018 v4.0 → 2019 v5.0 → 2020 v6.2 → 2021 v6.3





개발 배경

OpenFOAM의 부족한 점 보완

DIY CFD의 기본 프레임

공개 소스 패키지

→

OpenFOAM의 활성화

맞춤형 CFD의 확대

CFD의 대중화

BARAM의 구조 / 개발 방법

솔버의 안정성/정확성

- 좋지 않은 격자에서 안정성/정확성 문제
- 초기조건에 민감한 문제

복잡한 사용방법

- 많은 파일에서 경계조건 설정
- 복잡한 수치해석 기법 설정
- 복잡한 데이터 추출 / 모니터링 방법

불편한 사용자환경

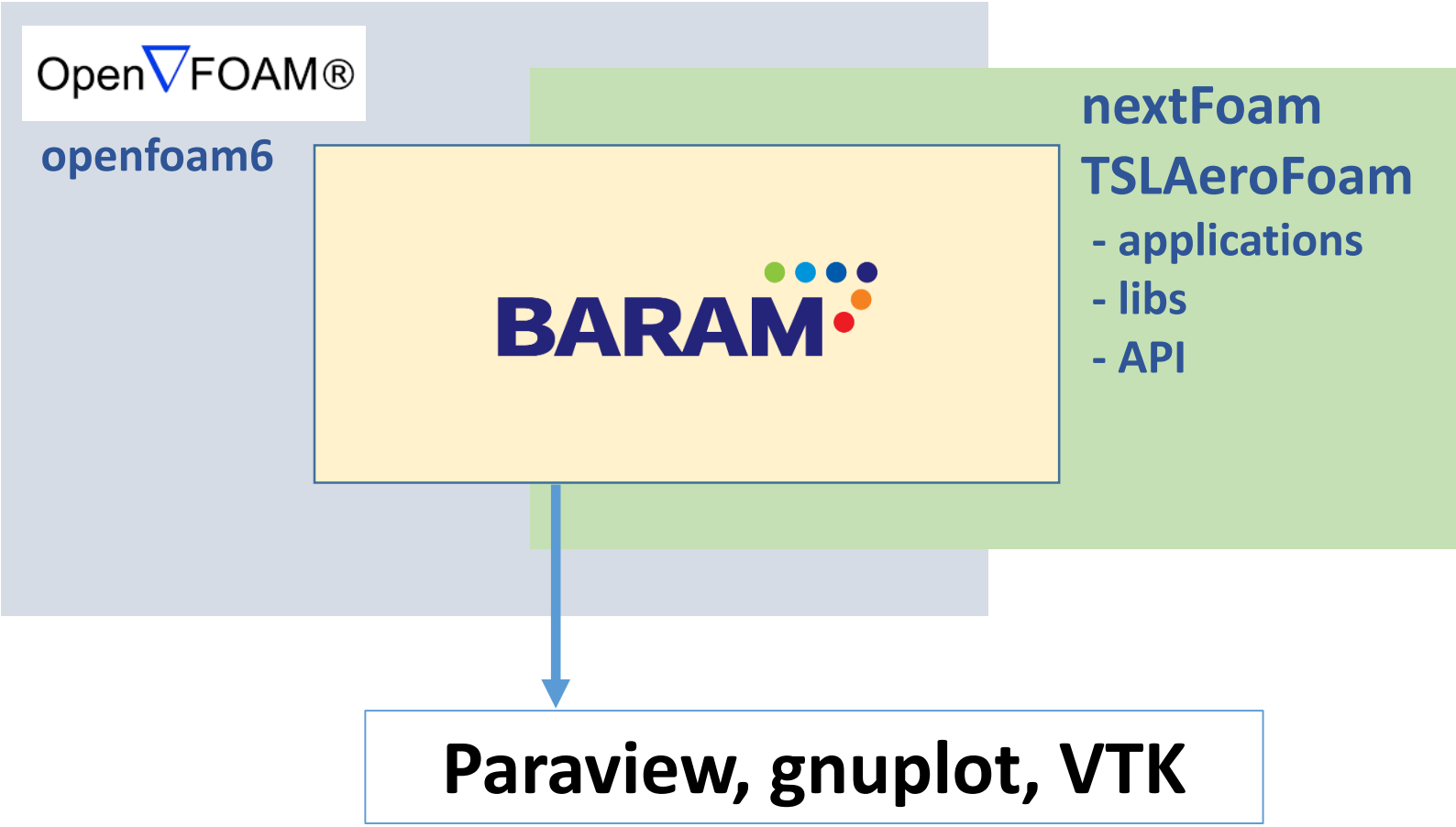
- Text User Interface
- 익숙하지 않은 OS, ,파일 편집기

OpenFOAM 코드 개선

API 개발

Graphic Interface 개발

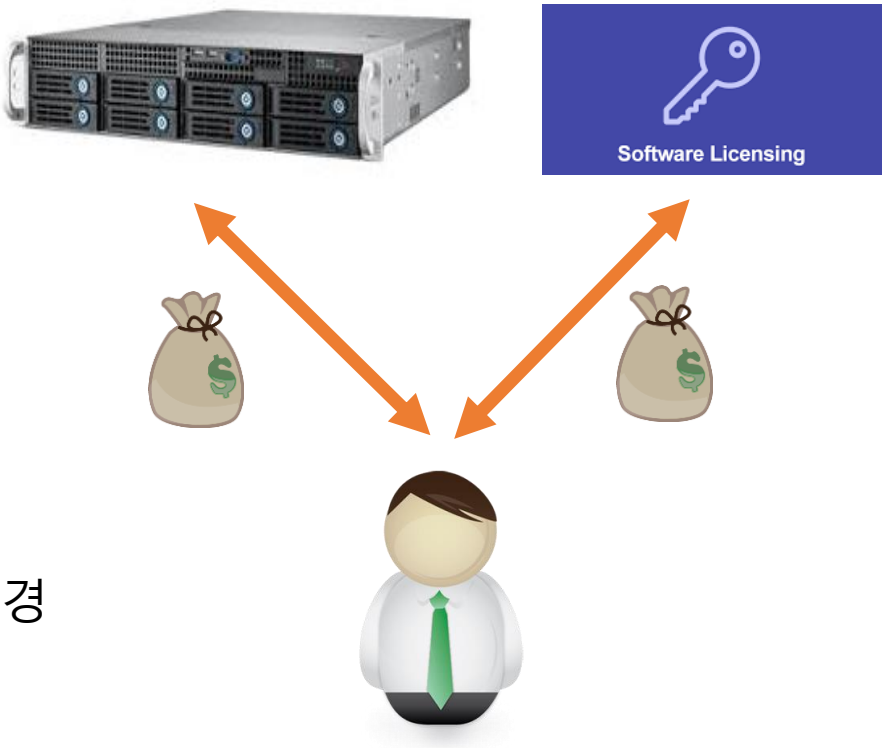
BARAM의 구조 / 개발 방법



개발 방향 – CFD 사용 환경의 변화

CFD 사용 방법의 Game Rule Change

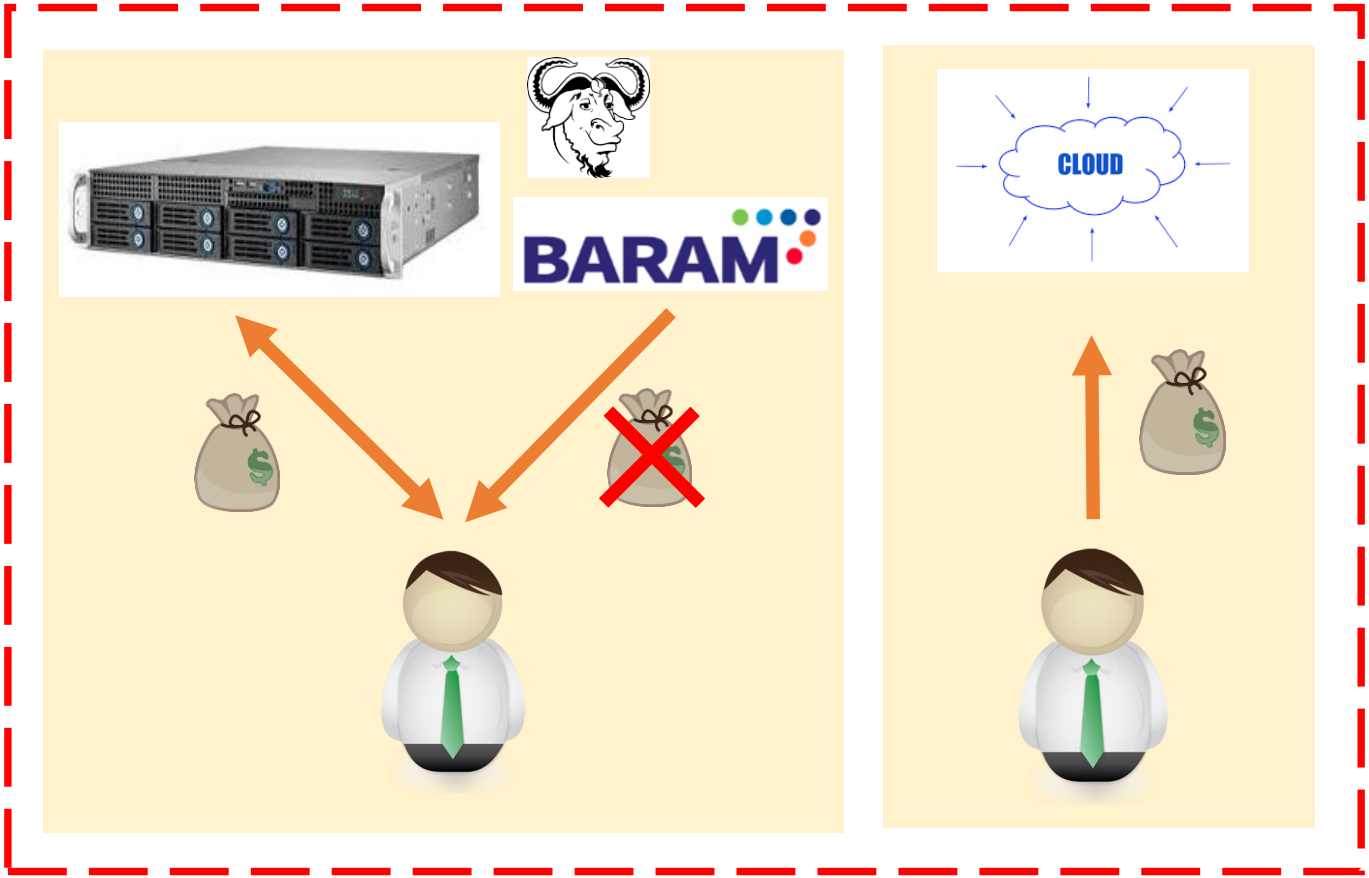
- 클라우드, 공개소스, 인공지능, 디지털트윈, 메타버스…
- **새로운 사용 방식, 상용 프로그램의 대안 요구**



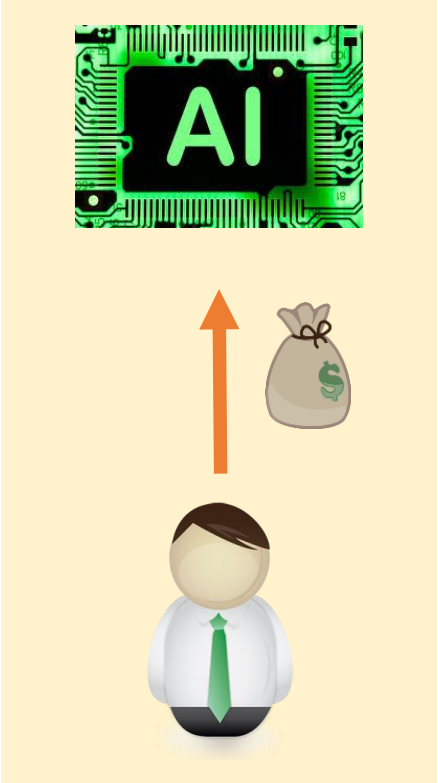
지금의 CFD 사용 환경

개발 방향 – CFD 사용 환경의 변화

CFD 사용 방법의 Game Rule Change

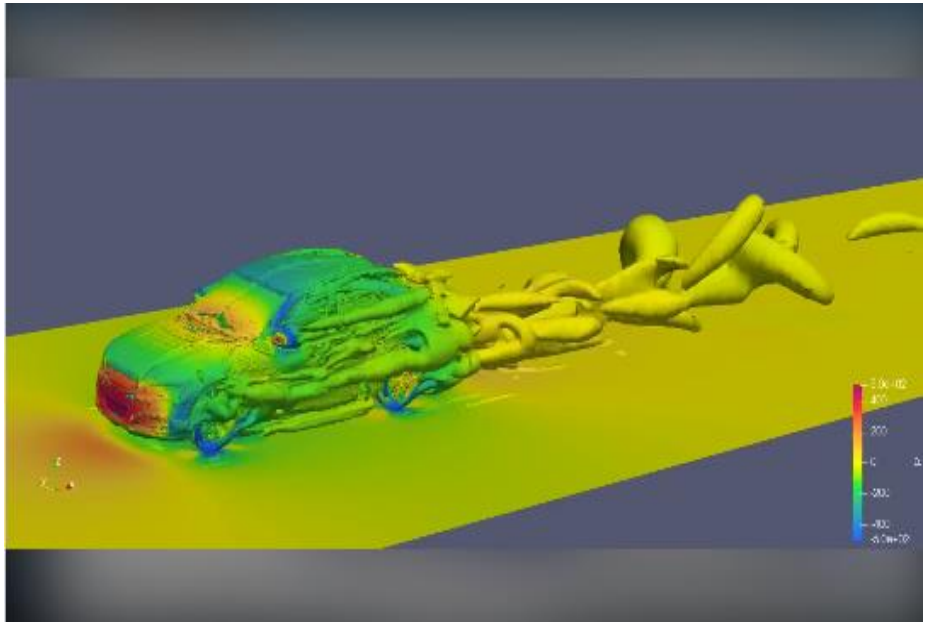


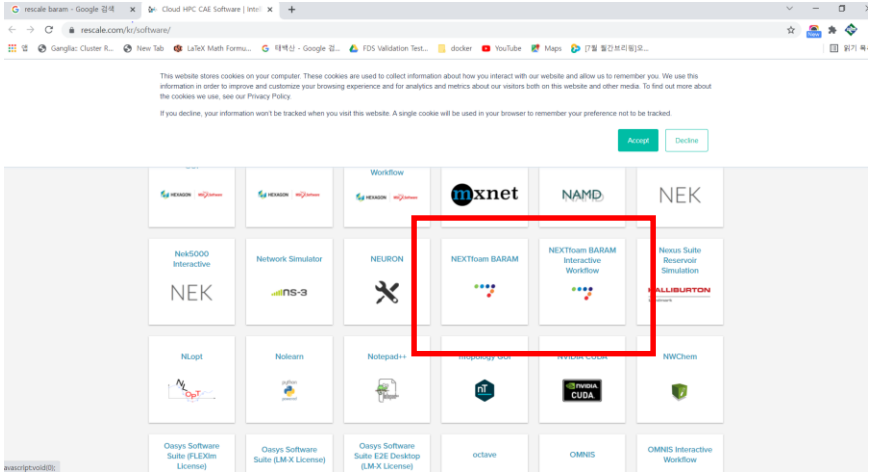
현재

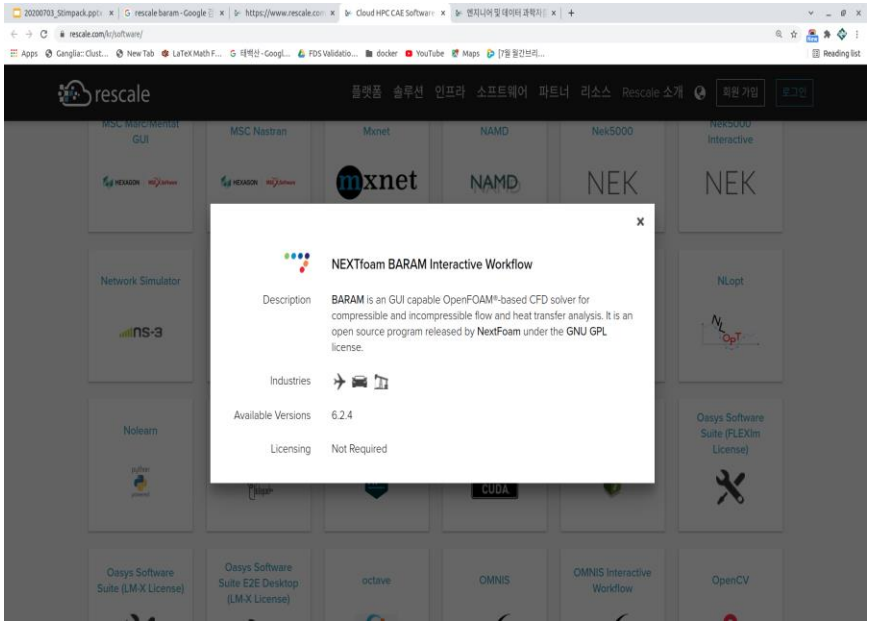


미래

BARAM in Rescale cloud







NEXTfoam
open source CFD consulting

v6.3 기능

- Solver
  - s(p)impleNFoam
  - buoyantS(P)impleNFoam
  - TSLAeroFoam
  - speciesS(P)impleNFoam
  - chtMultiRegionS(P)impleNFoam
  - PCNFoam
  - speciesPCNFoam
  - speciesTSLAeroFoam
- Passive scalar
- cellZone
  - MRF
  - porous
  - sliding mesh (비압축성)
  - fixedVelocity
- Mesh
  - snappyHexMesh
  - cfMesh
  - Mesh convert(fluent, starCCM+…)
  - Create baffle, cyclic
  - Check, scale, translate, rotate
  - refineWallLayer
  - Conver to axi-symmetric/2D mesh
- Source
  - Uniform source
  - Time dependent source (polynomial, piecewise linear, csv file)
  - fixedValue

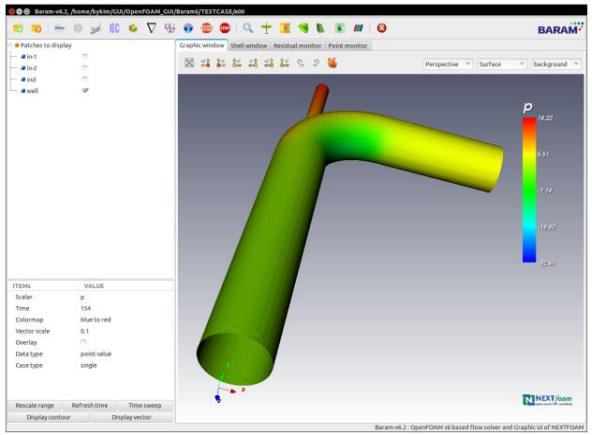
v6.3 기능

- Boundary conditions
  - velocityInlet
  - surfaceNormalVelocityInlet
  - pressureInlet
  - pressureOutlet, pressureOutletExt
  - adiabaticWall(noSlip, slip, movingWall, rotation, translate)
  - isothermalWall
  - heatFluxWall
  - convectionWall
  - thermoCoupledWall
  - Symmetry, emtpry, wedge, cyclic
  - ABLInlet, ABLWall
  - internalInterface
  - rotationalPeriodic
  - translationalPeriodic
  - porousJump
  - Fan
  - farfieldRiemann
  - subsonicInflow
  - subsonicOutflow
  - supersonicInflow
  - supersonicOutflow
  - Temporal profile
  - Spatial profile

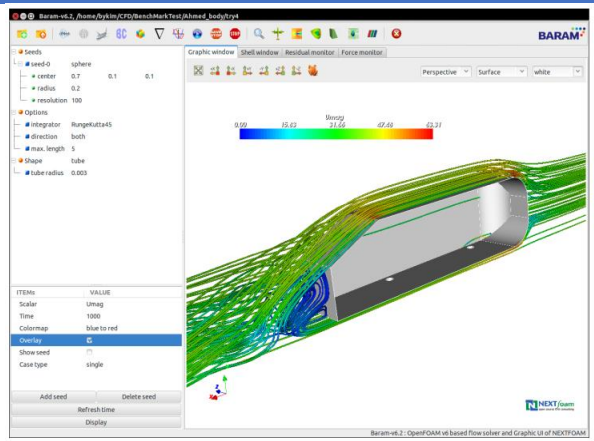
v6.3 기능

- Monitoring / data extract
  - Residual
  - Force
  - Point, surface, volume(average, integral, min, max, CoV)
- Graphic post-processing
  - Patch scalar
  - Cutting plane
  - Iso-surface
  - Clip
  - Streamline
- Create fields
  - Vorticity, Q, yPlus, wallShearStress, wallHeatFlux, totalPressure
- Initialize / run option
  - setFields, mapFields, potentialFoam
  - Passive scalar on/Off, scalar only
  - Set time to 0
  - Modify condition
  - Batch run(Mach, AOA, AOS, boundary value)
- Numerical schemes
  - Discretization scheme(first/second)
  - Relaxation factors
  - Convergence criterior
  - Pressure-velocity coupling
  - nOuterCorrectors
  - Sub-iteration(tol, relTol, min/maxIter)
  - nCorrectors
- Limit temperature

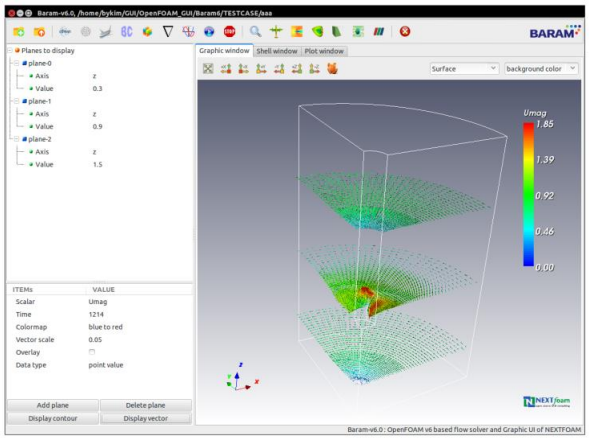
Tutorials



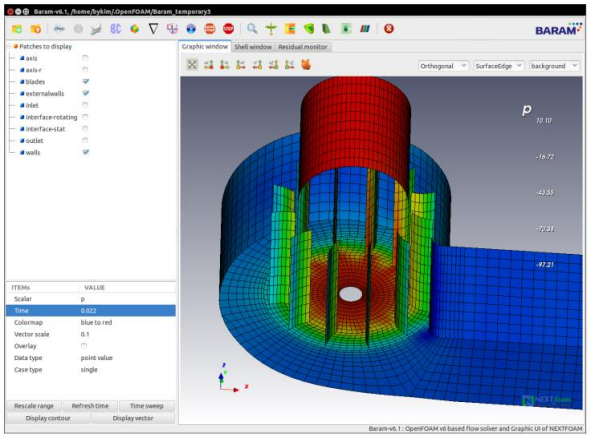
Mixing pipe
- cfMesh 격자생성
- 비압축성유동



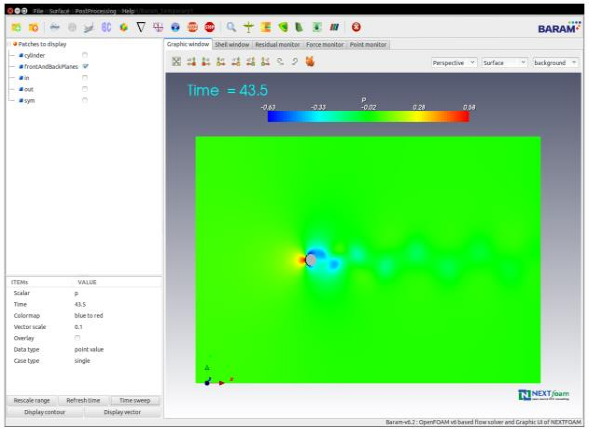
Ahmed body
- cfMesh 격자생성
- 비압축성 공력해석 검증



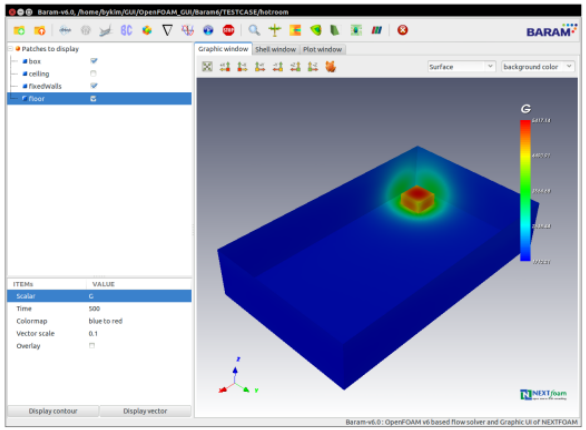
Mixer
- baffle, periodic B.C
- MRF



fan
- Sliding mesh

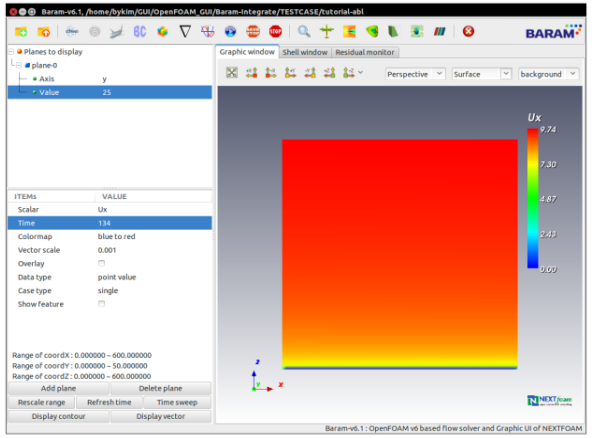


2d cylinder
- 2차원 계산, 모니터링
- Vortex shedding 검증

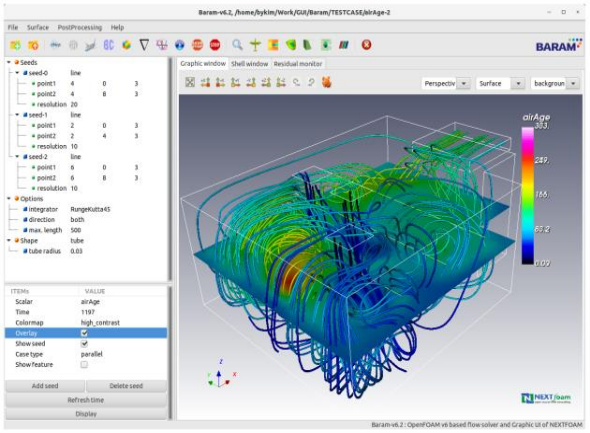


Hot room
- Natural convection
- Radiation

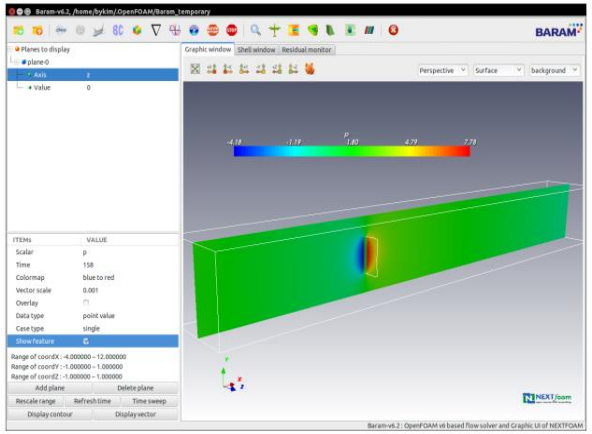
Tutorials



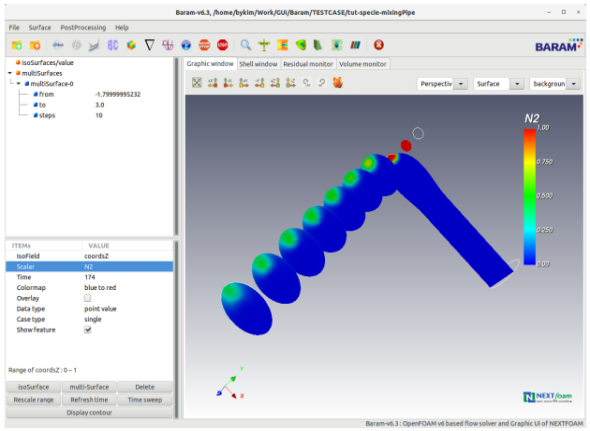
대기경계층
- 경계조건 검증



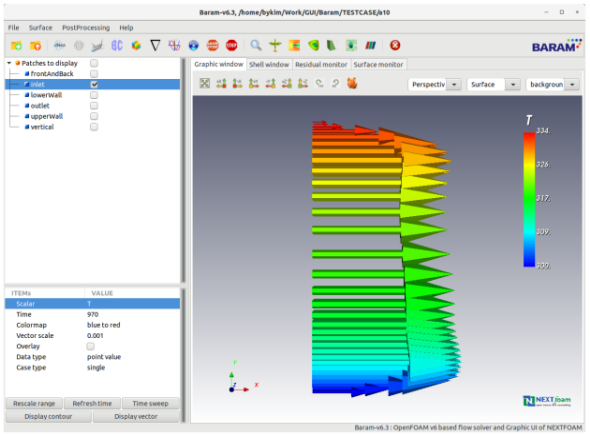
HVAC
- Passive scalar
- Air age



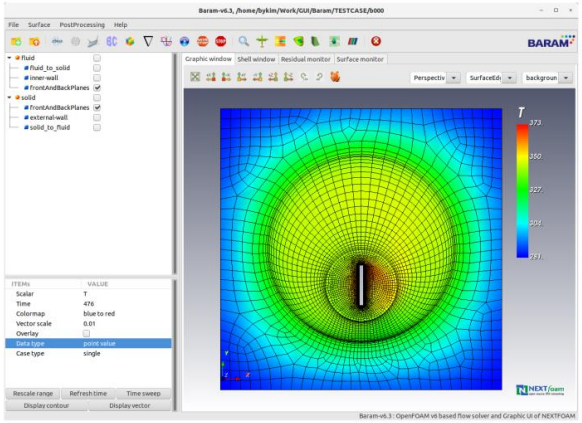
Duct flow
- Internal boundary
- Porous jump



Mixing pipe
- Species transport



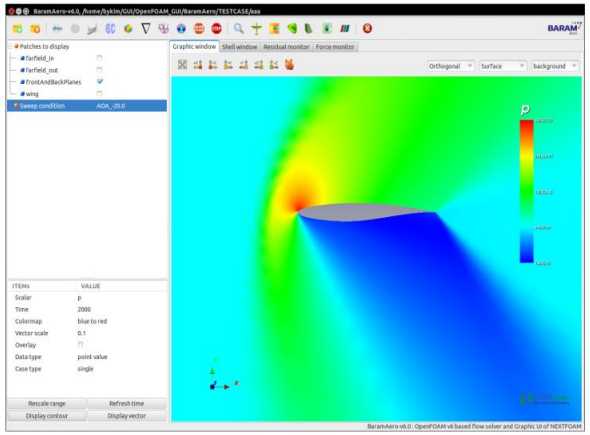
pitzDaily
- Profile B.C
- Time varying B.C



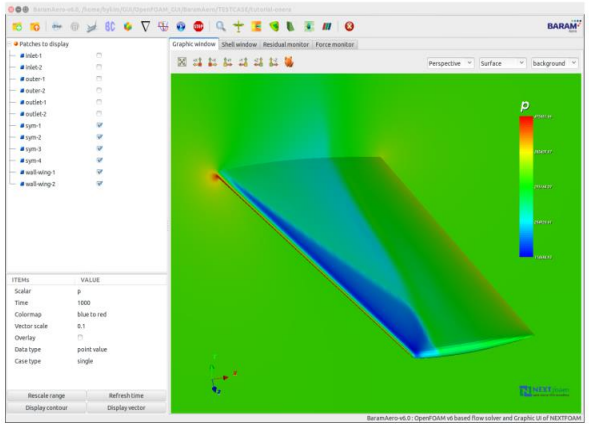
Hot bar
- Conjugate heat transfer
- multi-region

NEXTfoam
open source CFD consulting

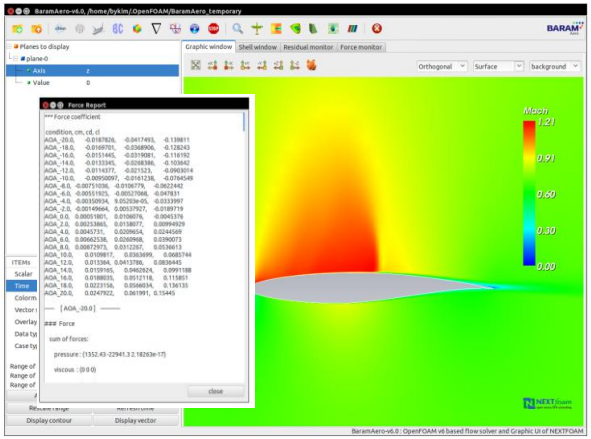
Tutorials



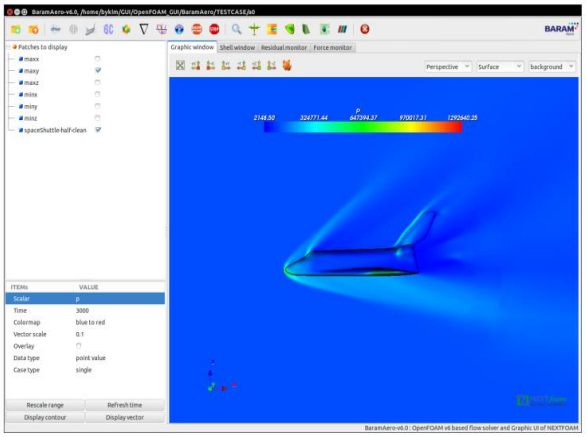
RAE2822 airfoil
- 압축성 유동 검증



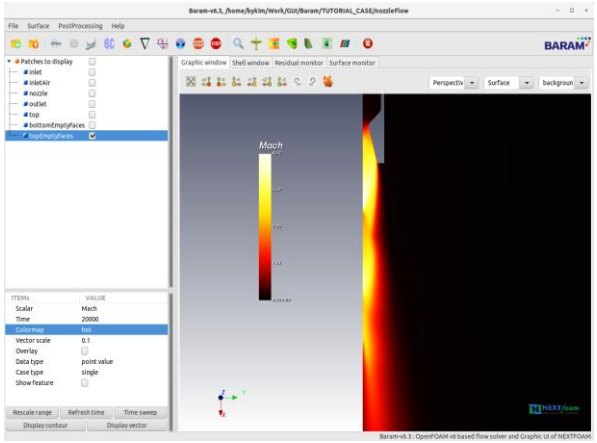
ONERA M6 wing
- 압축성 유동 검증



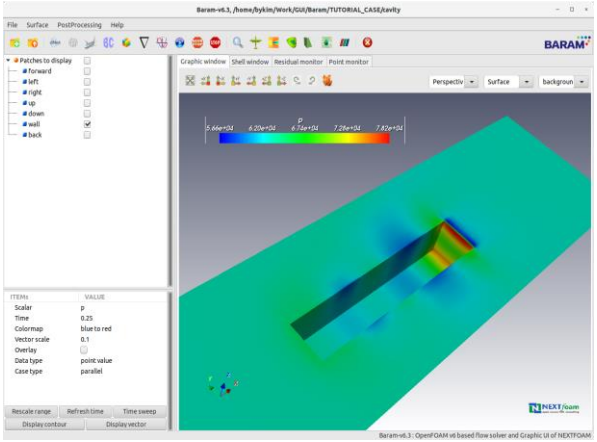
RAE2822 airfoil
- AOA sweep



Space shuttle
- cfMesh 격자생성
- Supersonic flow



Nozzle flow
- axi-symmetric mesh generation
- Supersonic flow



Cavity
- 천음속 공동유동 검증
- 압력기반 솔버(PCNFoam)

NEXTfoam
open source CFD consulting

앞으로의 계획

- CFD 코드 기능 확장
  - 물리 모델, 솔버 성능
  - 격자 관련 문제
- 사용자환경 개선
  - 안정성, 그래픽 속도, 서버-클라이언트, Windows
- 지원 서비스 시스템 개발
  - 설치, 교육, 기술지원, 매뉴얼
  - Validation case, tutorial case

앞으로의 계획

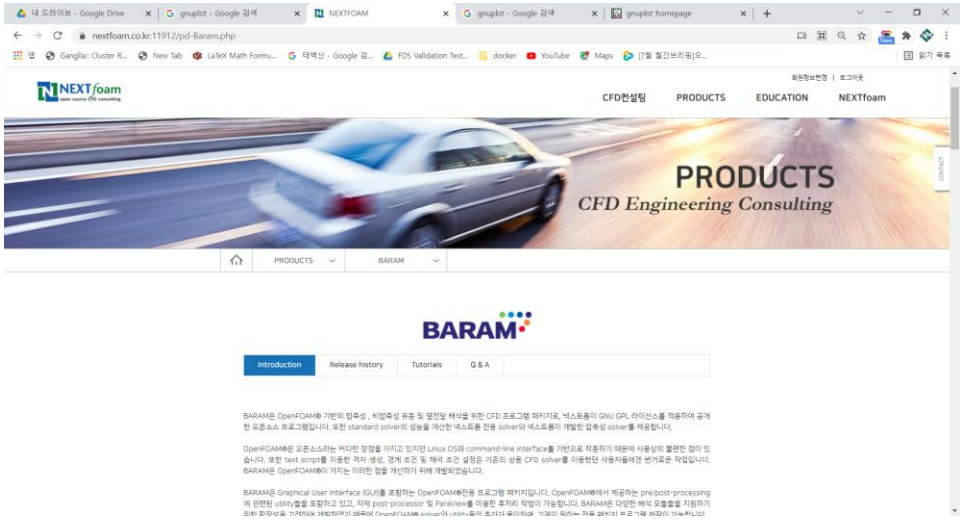
2022

- 다상유동
  - VOF
  - cavitation
- 압축성유동
  - transient TSLAeroFoam
- 설치 파일 문제
- Windows 문제

- 입자유동(DPM, spray)
- 난류(Low Re, RSM, DES/LES/DNS)
- 반응
- 격자 운동
- 솔버의 성능
- 격자 관련 문제
- 사용자환경 개선
- 매뉴얼/기술지원/교육…

감사합니다.

그리고…



https://nextfoam.co.kr:11912/pd-Baram.php

Species transport solver – 개요

- 화학 반응없이 화학종 혼합을 계산할 수 있는 솔버
- Standard solver는 reactingFoam, fireFoam 등의 연소 솔버가 있지만…
- 비현실적인 온도 분포
  - Vijaya Kumar의 modifiedReactingFoam 참조
- Diffusivity 설정 기능 부족
  - Turbulent Schmidt No., mass diffusivity 추가
- 정상상태 솔버가 없음
- 압축성 솔버가 없음

- 개발 솔버
  - species(P)impleNFoam, speciesPCNFoam, speciesTSLAeroFoam

Species transport solver – 개발 방법

1. speciesS(P)impleNFoam.C 코드 제작
   - buoyantS(P)impleNFoam 기반으로 수정
2. 화학종전달방정식 코드 작성(YEqn.H)
   - modifiedReactingFoam 의 YEqn.H 기반으로 수정
3. 에너지 방정식 수정
4. createFields 코드 수정

Species transport – 솔버

speciesSimpleNFoam.C

```
...
#include "fvCFD.H"
#include "turbulentFluidThermoModel.H"
#include "radiationModel.H"
#include "simpleControl.H"
#include "fvOptions.H"
#include "momentum.H"
#include "finiteVolumeFunctions.H"
#include "globalIOFunctions.H"
#include "constrainPhi.H"
#include "constrainPhig.H"

#include "rhoReactionThermo.H"
// * * * * * * * * * * * * * * * * * * * * * * * * * //

int main(int argc, char *argv[])
{
...
   #include "createFields.H"
   #include "createFieldRefs.H"
...
   while (simple.loop())
...
      #include "UEqn.H"
      #include "pEqn.H"
      turbulence->correct();

      #include "YEqn.H"
      #include “EEqn.H”

      #include “updateDensity.H”
...
      }
...
```

speciesPimpleNFoam.C

```
...
#include "fvCFD.H"
#include "rhoThermo.H"
#include "turbulentFluidThermoModel.H"
#include "radiationModel.H"
#include "pimpleControl.H"
#include "fvOptions.H"
#include “fvcSmooth.H”
#include "momentum.H"
#include "finiteVolumeFunctions.H"
#include "globalIOFunctions.H"
#include “localEulerDdtScheme.H”
#include "constrainPhi.H"
#include "constrainPhig.H"
#include "rhoReactionThermo.H"
// * * * * * * * * * * * * * * * * * * * * * * * * * //

int main(int argc, char *argv[])
{
...
   #include "createFields.H"
   #include "createFieldRefs.H"
...
         while (pimple.loop())
         {
...
            #include "UEqn.H"
            #include "YEqn.H"

            while (pimple.correct())
            {
               #include "pEqn.H"
...
```

Species transport – YEqn.H

- 지배방정식

$$\frac{\partial}{\partial t}(\rho Y_i)+\nabla\cdot(\rho\vec{v}\,Y_i)=-\nabla\cdot\vec{J}_i+S_i$$

J : diffusion flux, S : Source term

$$\vec{J}_i=-\left(\rho D_{m,i}+\frac{\mu_t}{Sc_t}\right)\nabla Y_i$$

$D_{m,i}$: mass diffusivity

Sc_t : turbulent Schmidt number

reactingFoam에는 없는 turbulent Schmidt No. 와mass diffusivity 고려

mvConvection->fvmDiv(phi,Yi) 사용하지 않음

```
{
  scalar Sct(diffusivities.lookupOrDefault("Sct",0.7));
  dimensionedScalar Dm(diffusivities.lookup("Dm"));

  volScalarField Yt(0.0*Y[0]);

  forAll(Y, i)
  {
    if (i != inertIndex && composition.active(i))
    {
      volScalarField& Yi = Y[i];

      fvScalarMatrix YiEqn
      (
        fvm::ddt(rho, Yi) // for PIMPLE
       //+ mvConvection->fvmDiv(phi, Yi)
       + fvm::div(phi, Yi)
        - fvm::laplacian(rho*Dm + turbulence->mut()/Sct, Yi)
       ==
        fvOptions(rho, Yi)
      );

      YiEqn.relax();

      fvOptions.constrain(YiEqn);

      YiEqn.solve(mesh.solver("Yi"));

      fvOptions.correct(Yi);

      Yi.max(0.0);
      Yt += Yi;
    }
  }

  Y[inertIndex] = scalar(1) - Yt;
  Y[inertIndex].max(0.0);
}
```

Species transport – EEqn.H

- 지배방정식

$$\frac{\partial}{\partial t}(\rho h)+\nabla\cdot(\rho\vec{v}h)+\frac{\partial}{\partial t}(\rho K)+\nabla\cdot(\rho\vec{v}K)=\frac{\partial p}{\partial t}+\rho\vec{v}\cdot\vec{g}-\nabla\cdot Q-\nabla\cdot\Sigma\vec{J}_i h_i$$

enthalpy transport due to inter-species diffusion

K : total kinetic energy

Q : conductive heat flux

J : diffusion flux

$$Q=\alpha_{eff}\nabla h+\alpha_{eff}(\Sigma h_i\nabla Y_i)$$

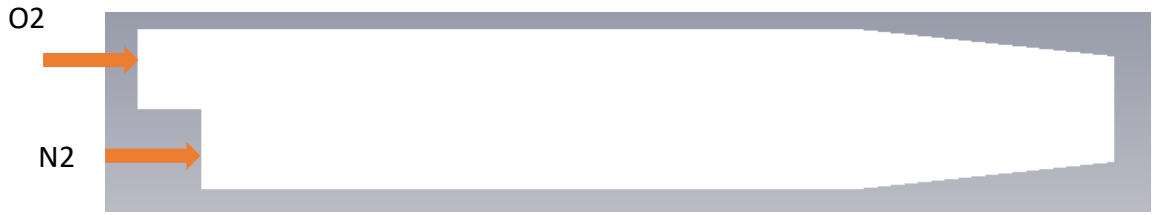
modifiedReactingFoam 에서 추가된 항
- reactinfFoam에는 없음
- 이 부분이 없으면 비현실적인 온도분포발생

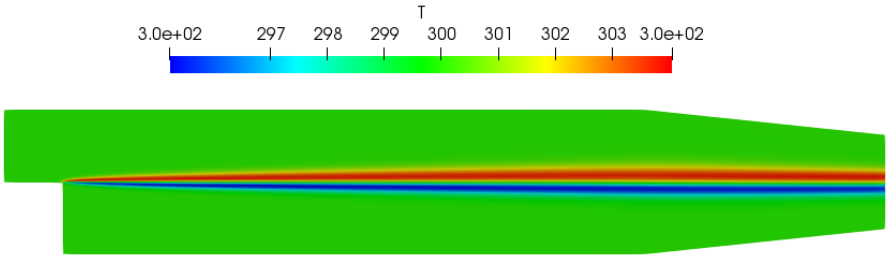
$$\vec{J}_i=-\left(\rho D_{m,i}+\frac{\mu_t}{Sc_t}\right)\nabla Y_i$$

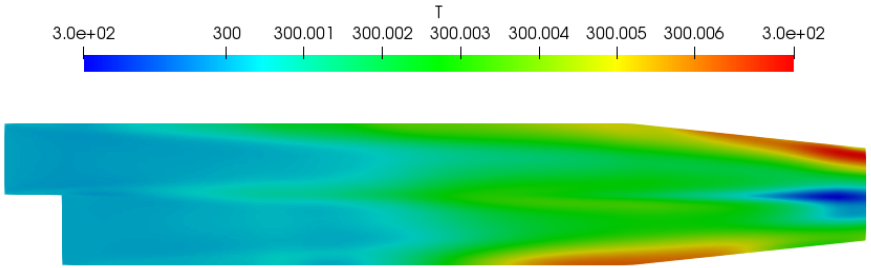
modifiedReactingFoam에서는 μ_{eff} 사용

EEqn.H

- 온도분포 검증
  - pitzDaily, steady, inlet velocity = 1m/s





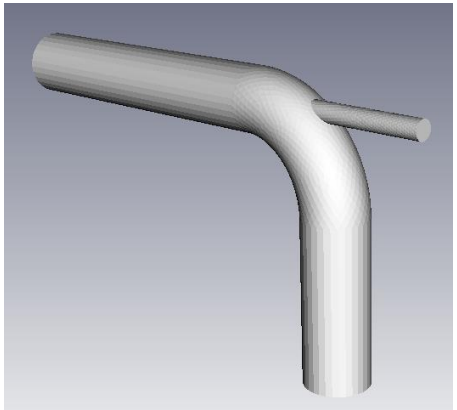


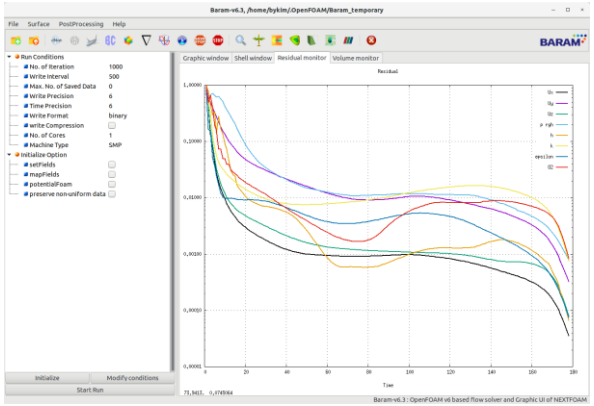
소수점 이하 자리수 는 limiter 때문에 생김

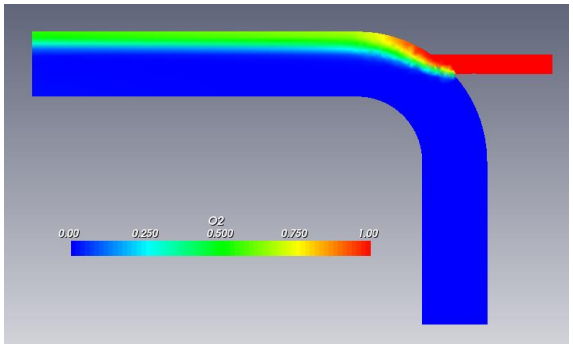
modifiedReactingFoam 수정 내용

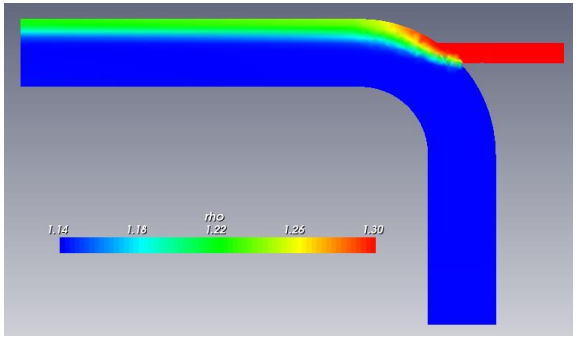
Test case 1

- in-1(아래쪽 입구)에는 N2, in-2(위쪽 좁은 입구)에는 O2가 1m/s으로 공급
- steady



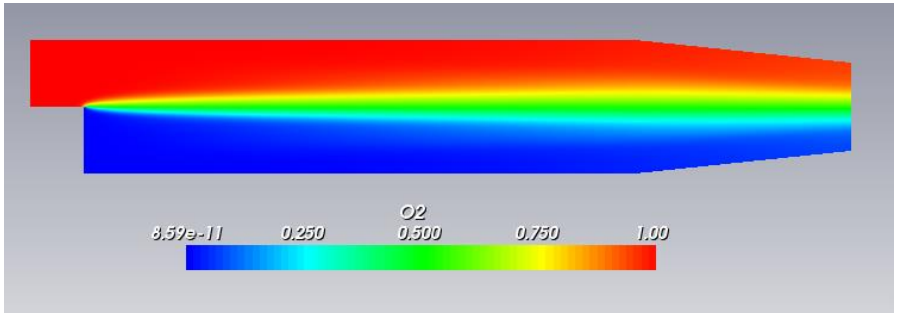




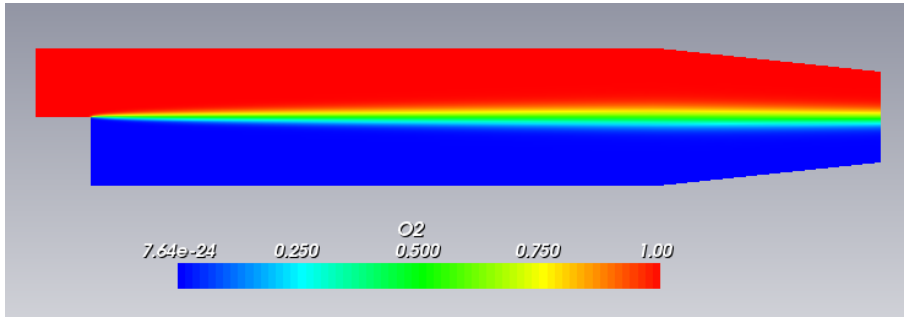


Test case 2

- pitzDaily, steady, inlet velocity = 1m/s
- Turbulent Schmidt No.의 영향 : 0.1 and 0.7



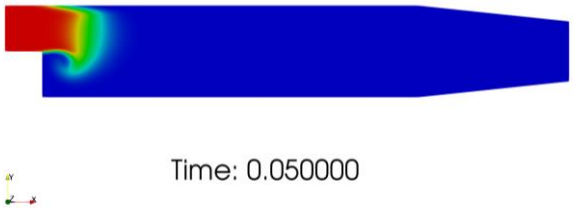
Sct = 0.1

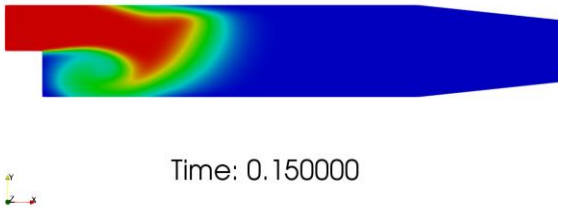


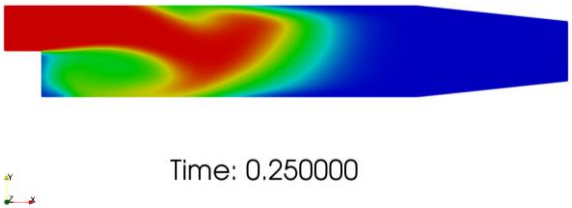
Sct = 0.7

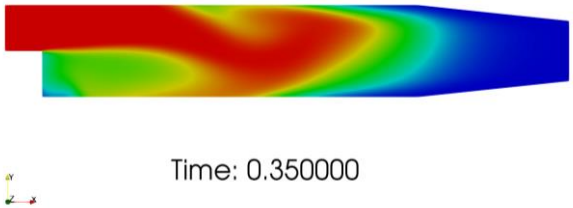
Test case 3

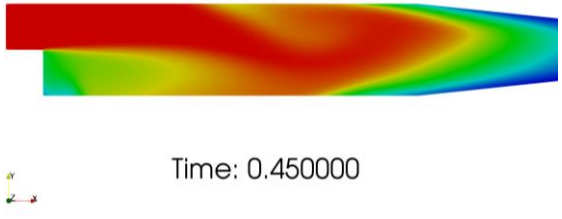
- pitzDaily, transient, inlet velocity = 1m/s
- 초기조건은 전체가 O2
- 입구에서 N2가 유입

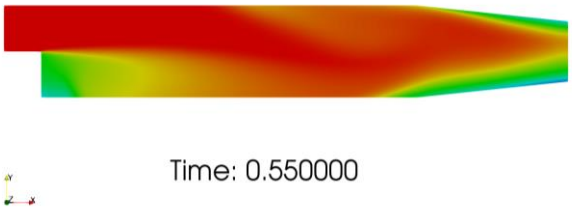












남은 일들

- YEqn.H에 thermal diffusion 추가

$$\vec{J}_i = -\left(\rho D_{i,m} + \frac{\mu_t}{S_{ct}}\right)\nabla Y_i - D_{T,i}\frac{\nabla T}{T}$$

- Mass diffusivity 보완
- Validation
- 압축성(speciesTSLAeroFoam)에서 에너지방정식 수정 사항 반영???