
はじめてのOpenFOAM®

その1

富山県立大学 中川慎二

オープンCAE勉強会@富山

2014年1月25日

Disclaimer: OPENFOAM® is a registered trade mark of OpenCFD Limited, the producer of the OpenFOAM software and owner of the OPENFOAM® and OpenCFD® trade marks. This offering is not approved or endorsed by OpenCFD Limited.

この講習会では、ユーザーガイドを参照しながら、作業を進めます。ユーザーガイドは下記サイトで入手可能です。

OpenFOAM® Documentation (オリジナル)

<http://www.openfoam.org/docs/>

<http://foam.sourceforge.net/docs/Guides-a4/UserGuide.pdf>

ソフトウェアマニュアル翻訳 (OpenCAE学会)

<http://www.opencae.jp/wiki/ソフトウェアマニュアル翻訳>

Disclaimer

OPENFOAM® is a registered trade mark of OpenCFD Limited, the producer of the OpenFOAM software and owner of the OPENFOAM® and OpenCFD® trade marks. This offering is not approved or endorsed by OpenCFD Limited.

コース概要

目的: OpenFOAMを利用し, 流動シミュレーションに必要な一連の作業を体験する。
OpenFOAMの基本的な使い方を学ぶ。

OpenFOAMマニュアル(ユーザーガイド)に掲載されている例題(チュートリアル)に, 実際にコンピュータを使って取り組む。

OpenFOAMでできそうなこと, できないことなど, 講習終了後の活用に向けた話題も取り上げる。

スケジュール

1. 使用システム説明
2. OpenFOAM概要
 - OpenFOAMとは?, 使用例紹介, ディレクトリ構造
3. 例題: キャビティ流れ
 - 概要, 格子生成, 条件設定, 流体解析, 可視化, 格子改造
4. 例題: ダムの崩壊
 - 概要, 格子生成, 条件設定, 流体解析, 可視化
5. さらにOpenFOAMを使うために
 - 情報元, 主なソルバー説明, 質疑応答

1. 講習会で使用する計算機の環境

ハードウェア
ノートPC

ソフトウェア

OS: Windows + VMware Player (仮想マシン実行環境)

ソフトウェア

OS: Linux Mint 16 Xfce 32bit版 (Ubuntu Linux 派生物)

<http://www.linuxmint.com/>

シミュレーションソフトウェア: OpenFOAM 2.2.2

<http://www.openfoam.com/>

動作環境の選択

- USBメモリ格納した仮想マシンをそのまま使う
 - ○ PCのドライブの空き容量が減らない。
 - × アクセスが やや遅い。
 - × 後で実行することができない。(コピーすれば可)
- USBメモリ内のフォルダをハードディスクにコピーして、使う。
 - 上記と逆。

動作環境の確認

- Windowsメニューから, Vmware Playerを起動
- Vmware Player で, 「仮想マシンを開く」を選択
- マイコンピュータから(USBメモリ)または(コピー先)を選び, LinuxMint16Xfce.vmxを開く
 - コピー or 移動 の質問があれば, コピーを選択
- 仮想マシンが起動

2. OpenFOAM概要

オープンソース・CFDソフトウェア “OpenFOAM”について

数値シミュレーションについて

CFD (Computational Fluid Dynamics): 数値
流体力学, 流体の数値シミュレーション

実現象 → 物理モデル → 数学モデル → シミュ
レーションモデル (様々な仮定)

作業の流れ:

プレ処理 → 計算 → ポスト処理

数値シミュレーションについて

作業の流れ:

ユーザマニュアル1章 p.17

プリ処理 → 計算 → ポスト処理

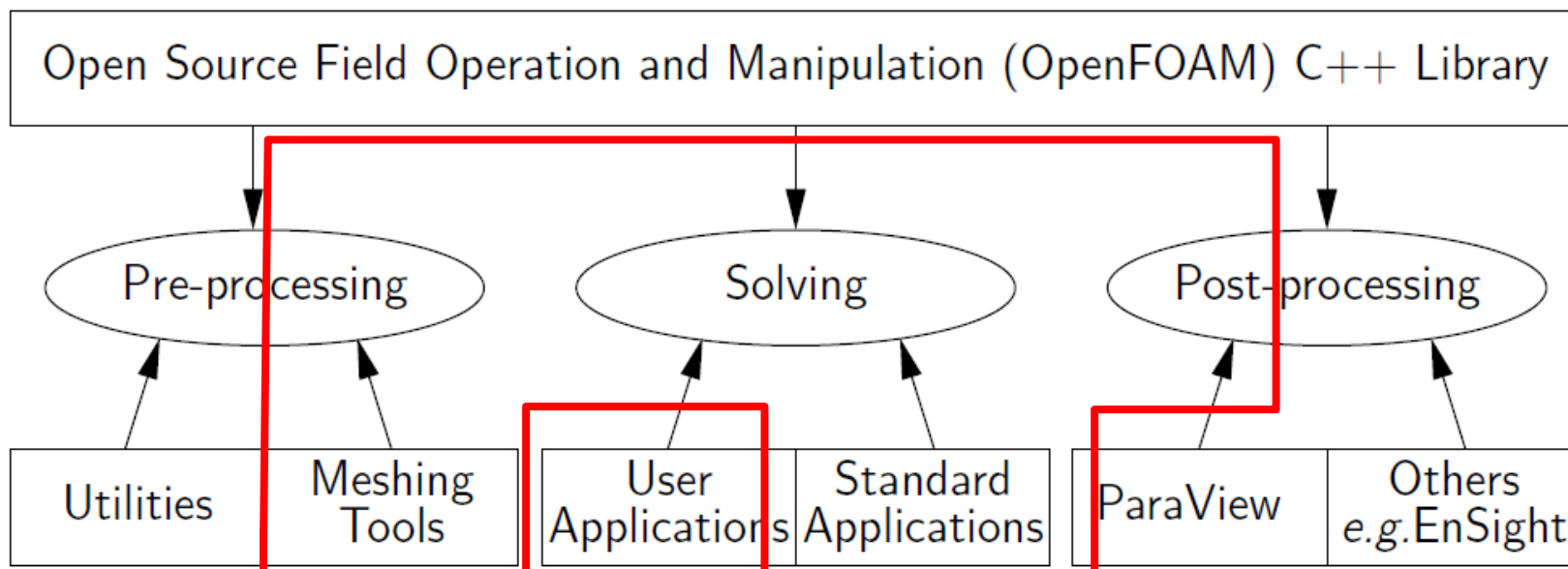


Figure 1.1: Overview of OpenFOAM structure.

OpenFOAMで利用するソフト

1. OpenFOAM
OpenFOAM本体, c++で記述されたプログラムの集まり。
各種ユーティリティを含む
2. paraview
オープンソースの可視化ソフトウェア
<http://www.paraview.org/>
Windows版, Linux版, MacOS版有り
3. メッシュ生成ソフトウェア
効率的な運用には 必要
4. gccなど
コンパイラ (GNU Compiler Collection)
ソースコードをコンパイルする場合には 必要

ディレクトリ構造 (OpenFOAM全体)

【インストール先】

/opt/openfoam222/ ←標準のインストール ディレクトリ

- ├ applications ←アプリケーションのソースファイル
- ├ bin ←実行ファイル
- ├ doc ←マニュアル
- ├ etc ←設定ファイル
- ├ lib ←ライブラリ
- ├ src ←各種部品のソースファイル
- ├ tutorials ←例題ファイル(オリジナル)
- └ wmake ←コンパイル関連(通常使用しません)

【作業フォルダ】

ユーザーのホームフォルダ(ディレクトリ)内に作成

ソースコード

- ファイルブラウザで下記ディレクトリに移動
/opt/openfoam222/applications/solvers/
incompressible/icoFoam
- メインプログラム
icoFoam.C

ソースコード

- プログラム

```
fvVectorMatrix UEqn (
    fvm::ddt(U) ← 時間微分項
    + fvm::div(phi, U) ← 対流項
    - fvm::laplacian(nu, U) ← 粘性項
);
solve(UEqn == -fvc::grad(p)); ← 圧力勾配項
```

- 基礎式

$$\frac{\partial \vec{U}}{\partial t} + \vec{U} \frac{\partial \vec{U}}{\partial \vec{X}} = -\frac{1}{\rho} \frac{\partial p}{\partial \vec{X}} + \nu (\nabla^2 \vec{U})$$

ディレクトリ構造 (ケース)

標準的な作業ディレクトリ構造

先のインストール先ではなく, 各ユーザの作業用ディレクトリ

\$HOME ←ユーザのホームディレクトリ: /home/user など

└ OpenFOAM

└ *user-2.2.2* ←ユーザの作業用ディレクトリ

└ run

└ tutorials ←例題作業ディレクトリ

└ incompressible ←非圧縮性流体ソルバ ディレクトリ

└ icoFoam ←icoFoamソルバー ディレクトリ

└ cavity ←ケース ディレクトリ

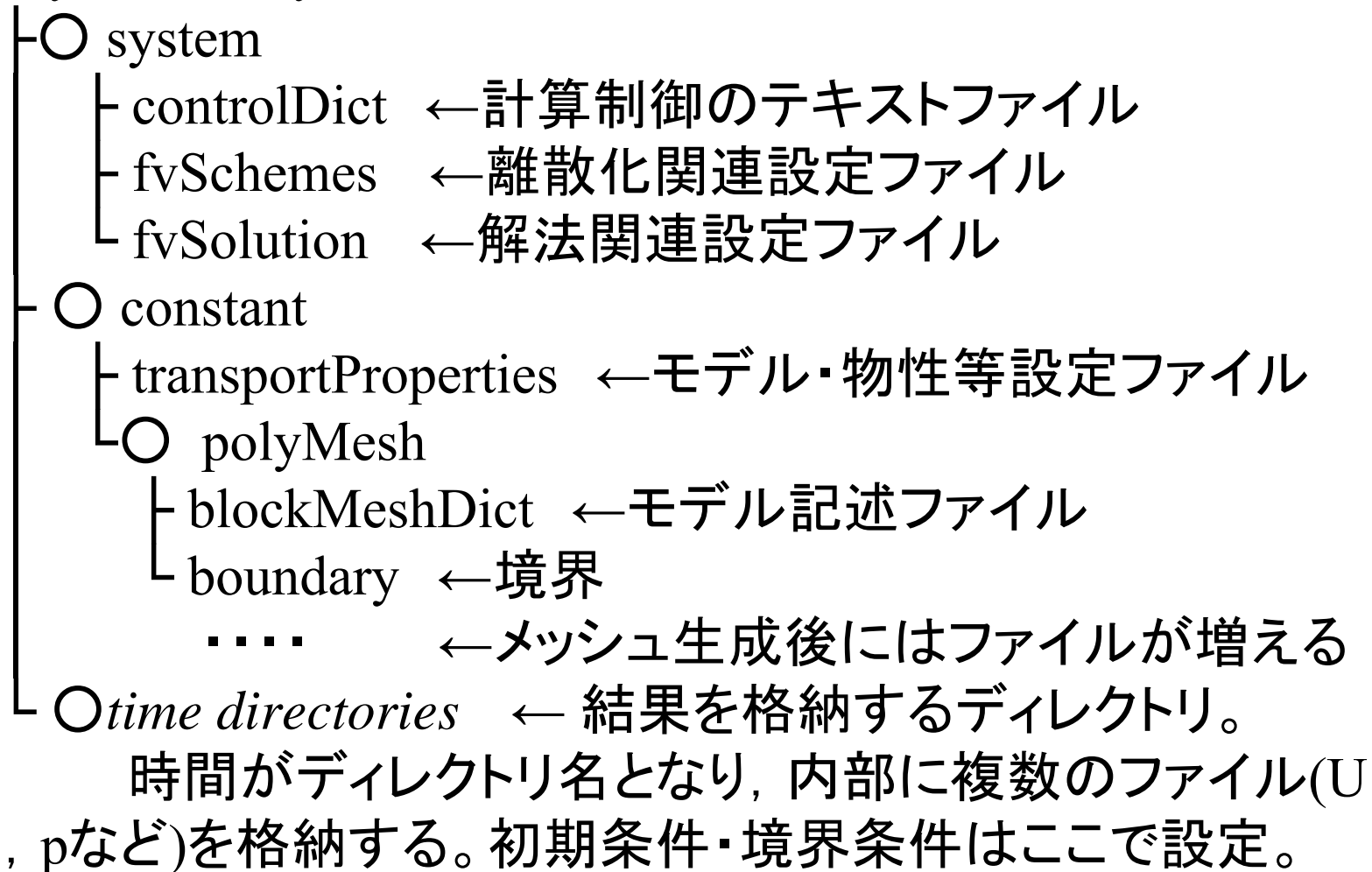
この中に, 各種計算条件を記載した

ファイルや, 計算結果が収納される。

詳細は次のスライド。

ディレクトリ構造 (ケース詳細)

cavity ← cavityケース ディレクトリ



詳細な説明は, ユーザマニュアル4.1節 p.101

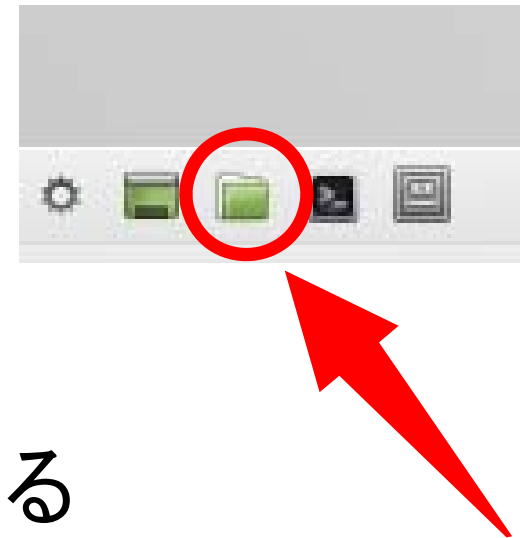
3. 例題

キャビティー流れ

作業：ディレクトリ構造の確認

ファイルマネージャーの起動

- 画面左下のフォルダアイコンをクリックし、Open Folderを選択する



- ホームディレクトリが表示される
 - `/home/user`
- 例題ディレクトリ(下記)へ移動する
 - `/home/user/OpenFOAM/user-2.2.2/run/tutorials`

作業：ディレクトリ構造の確認

- 例題ディレクトリ
 - /home/user/OpenFOAM/user-2.2.2/run/tutorials
- 問題の種類ごとにディレクトリが分かれている
- その中に、いろいろな場合に対する例題が納められている

Standard Solvers

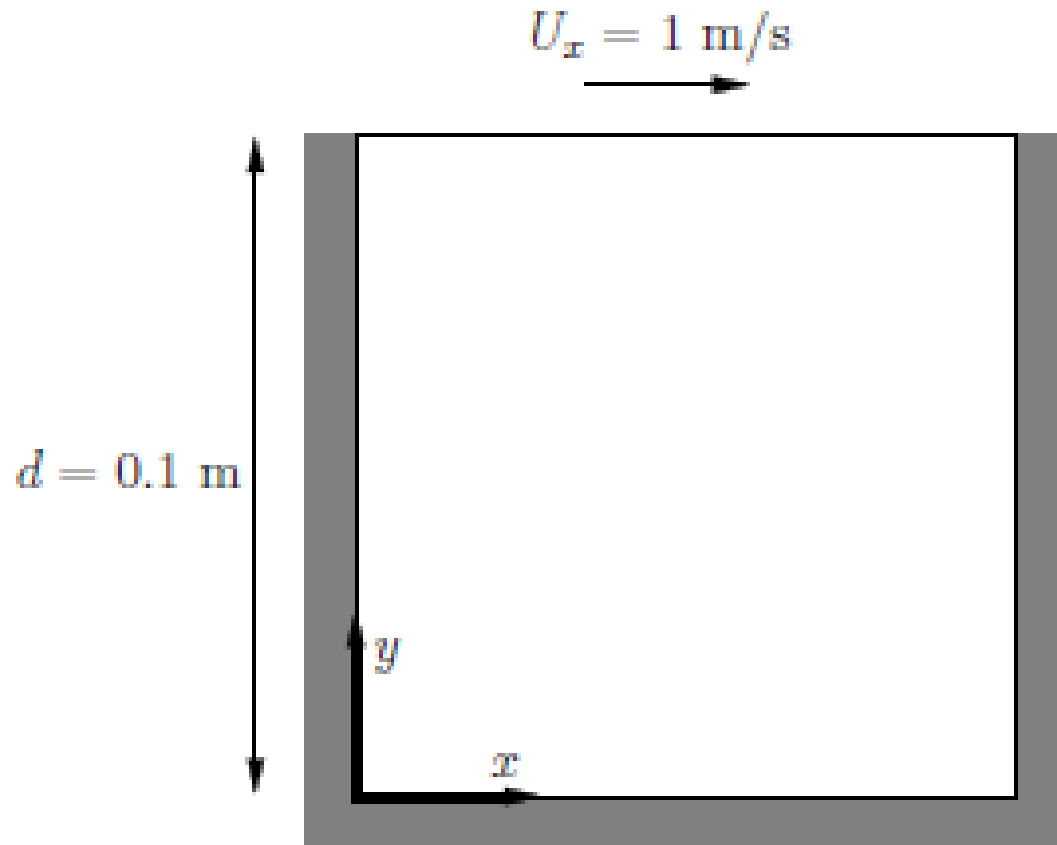
Incompressible flow

icoFoam

Transient solver for incompressible, laminar flow of
Newtonian fluids

例題1: キャビティ流れ

ユーザマニュアル2.1節 p.19



正方形容器

上蓋が移動

容器内に**非圧縮性**流体
(incompressible fluid)

2次元流, **層流**, **一定温度**

速度と圧力を求める

Figure 2.1: Geometry of the lid driven cavity.

Standard Application の
icoFoamを利用する

モデルの幾何形状

ユーザマニュアル2.1.1.1節 p.21

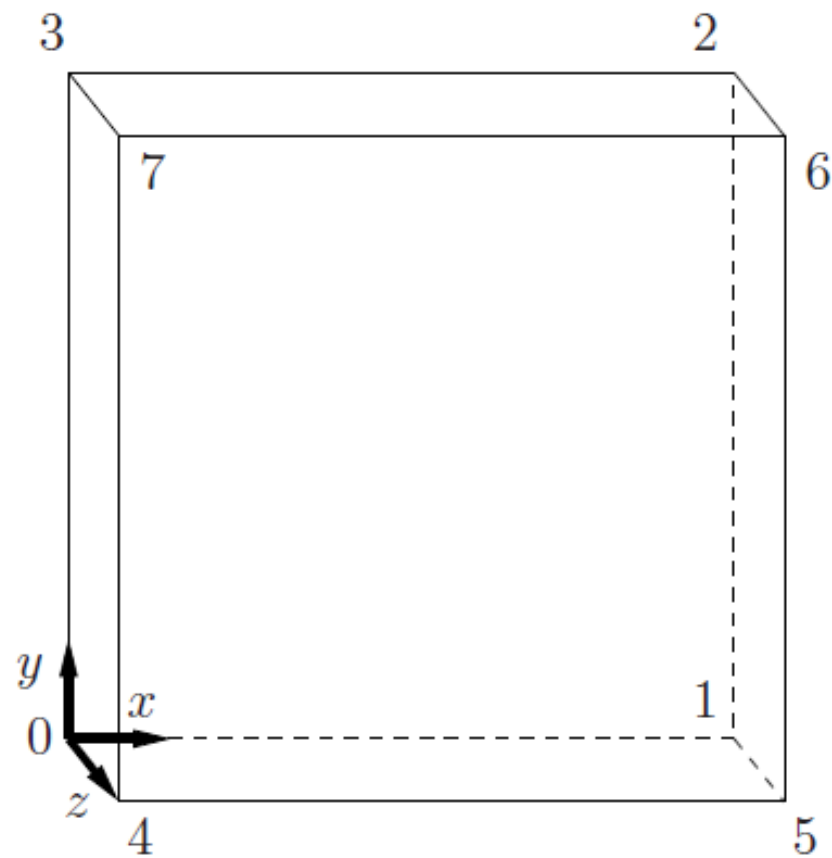


Figure 2.2: Block structure of the mesh for the cavity.

プリ処理 Pre-processing

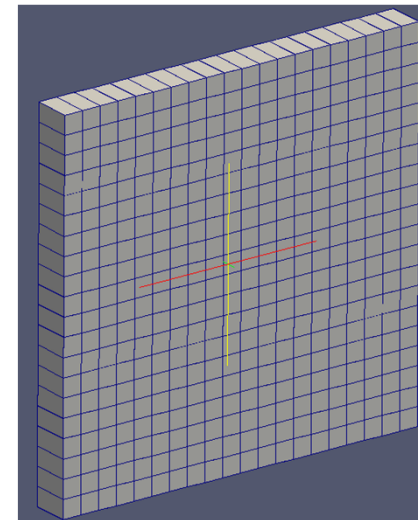
ユーザマニュアル2.1.1節 p.20

作業内容

- メッシュ生成 (Mesh generation)
- 境界条件と初期条件設定 (Boundary and initial conditions)
- 物性値設定 (Physical properties)
- 計算制御設定 (Control)
- 離散化と行列解法の設定 (Discretisation and linear-solver settings)

メッシュ

- 計算領域を多くの小さな領域に分割する
- 小領域をセルという
- 分割線をメッシュという
- シミュレーションでは, 各セルでの物理量を予測する



メッシュ生成

単純なメッシュ

- OpenFOAMで作成できる
- blockMeshDictというファイルにメッシュの生成方法を記述 → メッシュ生成コマンド blockMeshを実行

少し複雑なメッシュ

- OpenFOAMで作成できる
- 任意形状のSTLファイルに適合したメッシュの生成 → snappyHexMesh ユーティリティ

【作業：ファイルマネージャ】

- ファイルマネージャで,
/home/user/OpenFOAM/user-2.2.2/run
/tutorials/incompressible/icoFoam/cavity
/constant/polyMesh
まで移動し, blockMeshDictをダブルクリック
して開く。

blockMeshディクショナリの基本構造

ユーザマニュアル2.1.1.1節 p.21

blockMeshDictディクショナリの基本構造

convertToMeters 単位変換の係数

基本単位はm

たとえば, mm単位で記入するとき, この係数を0.001とする

vertices 節点 座標を与える

blocks ブロック

patches 面に関する情報(境界条件)

メッシュ作成指令書: blockMeshDict

convertToMeters 0.1; ← これから書く数字を, 0.1倍すると, 単位がmになる

vertices ← 節点

(

(0 0 0) ← 0から6までの各点の座標。この値を0.1倍するとm単位になる。

(1 0 0) ← この場合, x座標が $1 \times 0.1 = 0.1\text{m}$ の位置に点を置く。

(1 1 0)

(0 1 0)

(0 0 0.1)

(1 0 0.1)

(1 1 0.1)

(0 1 0.1)

);

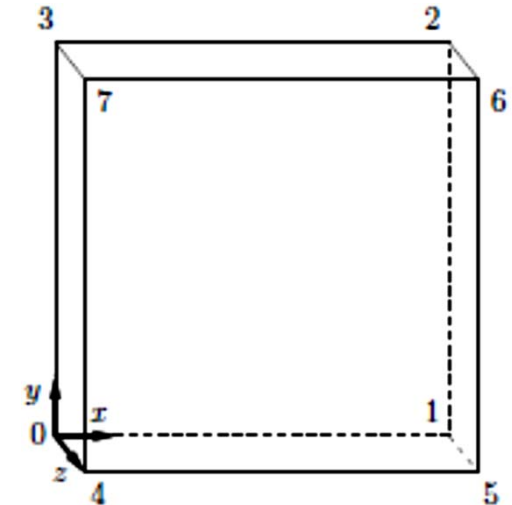
blocks ← ブロック(直方体(hex), 節点番号で指定する)

(

hex (0 1 2 3 4 5 6 7) (20 20 1) simpleGrading (1 1 1)

← 節点0~6で直方体を作る。x, y, z方向に20, 20, 1分割(等間隔)。

);



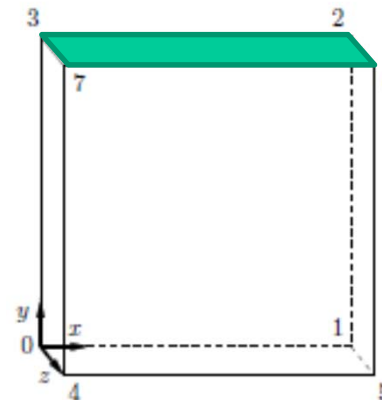
メッシュ作成指令書: blockMeshDict

boundary ← 同じ境界条件をまとめる

ユーザマニュアル2.1.1.1節 p.21

```
(  
  movingWall ← 境界条件に名前をつける:一定の速度で動く壁面  
  {  
    type wall; ← タイプを壁面に設定する  
    faces ( (3 7 6 2) ); ← 4つの節点で構成される面  
  }  
  fixedWalls ← 境界条件に名前をつける:固定された壁(速度は0)  
  {  
    type wall; ← タイプを壁面に設定する  
    faces ← 4つの節点で構成される面 が3個  
    ( (0 4 7 3) (2 6 5 1) (1 5 4 0) );  
  }  
  frontAndBack ← 境界条件:対称面(物理量の勾配は0)  
  {  
    type empty; ← タイプをemptyに設定する(2次元計算のため)  
    faces ← 4つの節点で構成される面 が2個  
    ( (0 3 2 1) (4 5 6 7) );  
  }  
);
```

movingWall



【作業: 端末】メッシュ生成: blockMeshの実行

ユーザマニュアル2.1.1.1節 p.22

- ファイルマネージャーで,

`/home/user/OpenFOAM/user-2.2.2/run`

`/tutorials/incompressible/icoFoam/cavity/`を選択,
右クリックして, 「Open in Terminal」をクリック。

- Terminal(端末)で, 下記コマンドを実行し, メッシュ生成ユーティリティblockMeshを実行する。

`blockMesh`

- 端末に, 実行結果が表示される。エラーメッセージが表示されていないか, 確認する。

境界条件および初期条件

- 時刻 0 のディレクトリに、初期条件が記述されたファイル (p と U) を用意する。

【作業: ファイルマネージャ】

現在のケースディレクトリ (/home/user/OpenFOAM/user-2.2.2/run/tutorials/incompressible/icoFoam/cavity) の下にある 0 ディレクトリまで移動し、ファイル p をダブルクリックして開く。

/0/pファイルの読み方

```
dimensions      [0 2 -2 0 0 0 0];
```

Kg **m** **s**



```
internalField   uniform 0;
```

```
boundaryField
```

```
{
  movingWall
  {
    type          zeroGradient;
  }
  fixedWalls
  {
    type          zeroGradient;
  }
  frontAndBack
  {
    type          empty;
  }
}
```

この変数 p の単位は m^2/s^2

(圧力／密度となる)

内部の値は 一様で 0

境界条件

movingWall, fixedWallsと
いう名の境界では、境界面
に垂直な方向の圧力勾配
は 0。

frontAndBack境界では、
empty → 2次元流れの境界

/O/Uファイルの読み方

```
dimensions      [0 1 -1 0 0 0 0];
internalField   uniform (0 0 0);
boundaryField
{
  movingWall
  {
    type          fixedValue;
    value         uniform (1 0 0);
  }
  fixedWalls
  {
    type          fixedValue;
    value         uniform (0 0 0);
  }
  frontAndBack
  {
    type          empty;
  }
}
```

Kg m s

[0 1 -1 0 0 0 0];

この変数 U の単位は m/s

ベクトルなので3成分をもつ

内部の値は 一様で(0,0,0)

境界条件

movingWall境界では、速度は一定値。x方向に1m/s

fixedWalls境界では、速度は一定値 0。静止。

frontAndBack境界では、empty → 2次元流れの境界

物性値設定

ユーザマニュアル2.1.1.3節 p.23

- ~Propertiesという名のディクショナリに物性値を記録(ケース・ディレクトリ/constant/に保存)
- icoFoamでは, 動粘度 ν をtransportPropertiesディクショナリで与える。
- 今回は $\nu = 0.01 \text{ m}^2\text{s}^{-1}$ と設定
 - レイノルズ数 $Re = U d / \nu = 1 \times 0.1 / 0.01 = 10$

【作業:ファイルマネージャ】

/home/user/OpenFOAM/user-2.2.2/run/tutorials/incompressible/icoFoam/cavity/constant

の下にある transportPropertiesファイル をダブルクリックして開き, 内容を確認。

計算制御設定

ユーザマニュアル2.1.1.4節 p.24

- 計算時間, データ読み込み, 書き出し制御等に関する情報は controlDict ディクショナリに記録
- controlDict ディクショナリは, ケースディレクトリ/systemディレクトリに保存する

【作業: ファイルマネージャ】

現在のケースディレクトリ

/home/user/OpenFOAM/user-2.2.2/run/tutorials/incompressible/icoFoam/cavity

の下にある system ディレクトリまで移動し, ファイル controlDict をダブルクリックして開く。

controlDict の読み方 (1)

ユーザマニュアル2.1.1.4節 p.24

<code>application</code>	<code>icoFoam;</code>	ソルバは <code>icoFoam</code>
<code>startFrom</code>	<code>startTime;</code>	計算を <code>startTime</code> 欄で指定して時刻から始める。(今回は0秒)
<code>startTime</code>	<code>0;</code>	
<code>stopAt</code>	<code>endTime;</code>	計算を <code>endTime</code> 欄で指定した時刻で止める。(今回は0.5秒)
<code>endTime</code>	<code>0.5;</code>	
<code>deltaT</code>	<code>0.005;</code>	時間の刻み幅を指定する。
<code>writeControl</code>	<code>timeStep;</code>	結果ファイルの書出し時刻を指定する。 <code>writeInterval</code> で指定した回数ごとに書き出す。(今回は $0.005 \text{秒} \times 20 \text{回} = 0.1 \text{秒}$ 毎)
<code>writeInterval</code>	<code>20;</code>	

より詳細な説明(他の選択肢など)は、ユーザマニュアル4.3節 p.108

controlDict の読み方 (2)

ユーザマニュアル2.1.1.4節 p.24

<code>purgeWrite</code>	<code>0;</code>	書き出しファイル数を制限するか。(今回は制限しない。)
<code>writeFormat</code>	<code>ascii;</code>	書き出しファイルをASCII形式に
<code>writePrecision</code>	<code>6;</code>	書き出すデータの有効桁数
<code>writeCompression</code>	<code>off;</code>	書き出しファイルの圧縮/非圧縮
<code>timeFormat</code>	<code>general;</code>	書き出しディレクトリの名前の付
<code>timePrecision</code>	<code>6;</code>	け方と桁数
<code>runTimeModifiable</code>	<code>true;</code>	各タイムステップの開始時に、 各種ディクショナリを再読み込み するかどうか

より詳細な説明は、ユーザマニュアル4.3節 p.109

離散化と行列解法の設定

ユーザマニュアル2.1.1.5節 p.25

有限体積法での離散化方法

- ケース/system/fvSchemesディクショナリ

行列解法、トレランス、アルゴリズム設定など

- ケース/system/fvSolutionディクショナリ

メッシュの確認

ユーザマニュアル2.1.2節 p.26

ポスト処理ソフトParaViewを使って、メッシュを確認する

【作業：端末】

- ケース「cavity」ディレクトリにいることを確認するため、下記コマンドを実行する。

pwd

- /home/user/OpenFOAM/user-2.2.2/run/tutorials/incompressible/icoFoam/cavityと表示されればよい。違う場所にいるときは、下記コマンドを実行する。

```
cd $FOAM_RUN/tutorials/incompressible/icoFoam/cavity
```

- 下記のコマンドを実行する

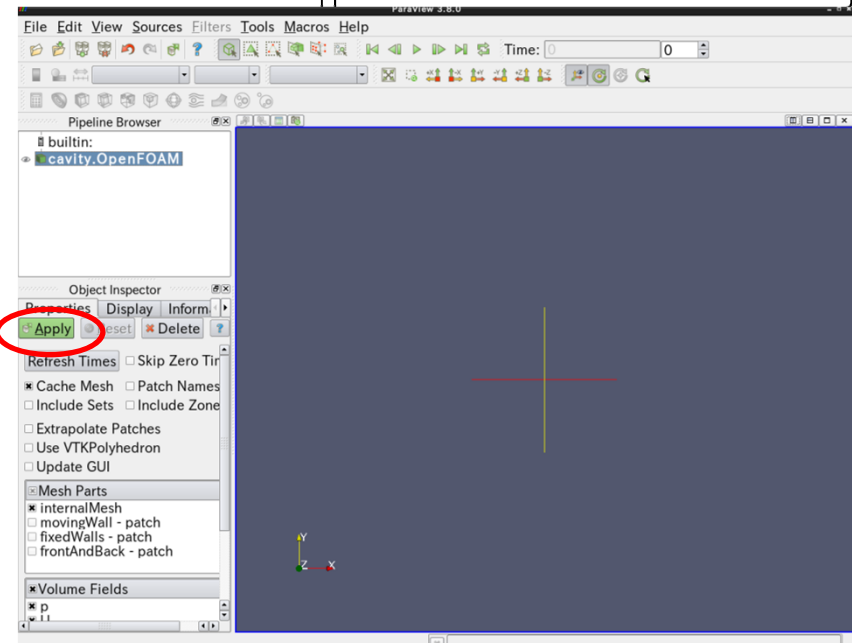
paraFoam

- ParaViewが起動する

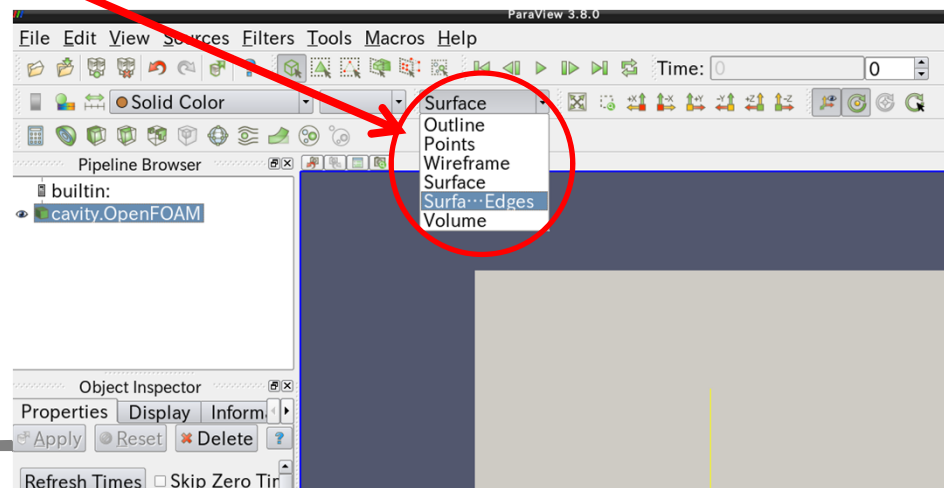
メッシュの確認

ユーザマニュアル2.1.2節 p.26

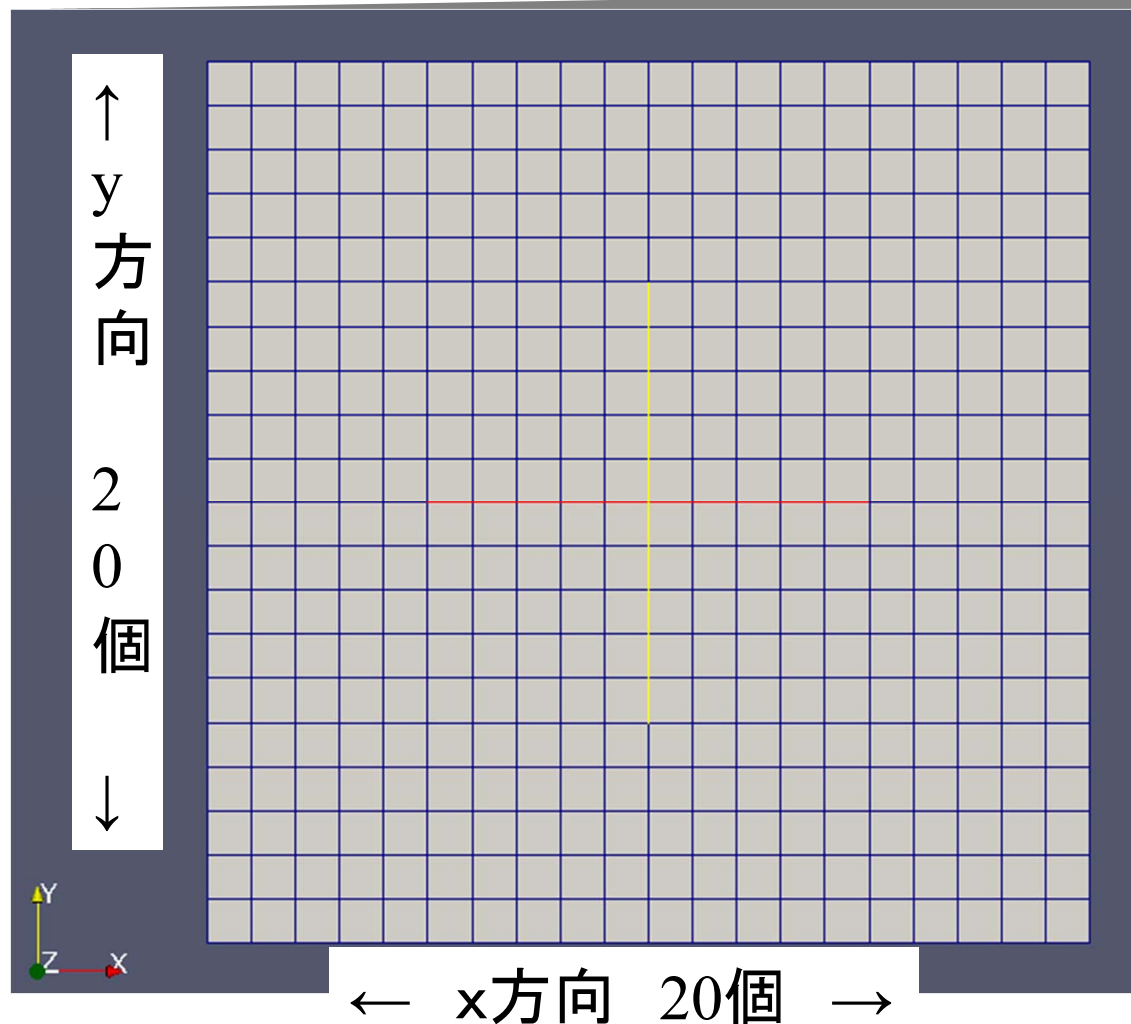
- ParaViewが起動する
- Applyボタン(緑)を押す



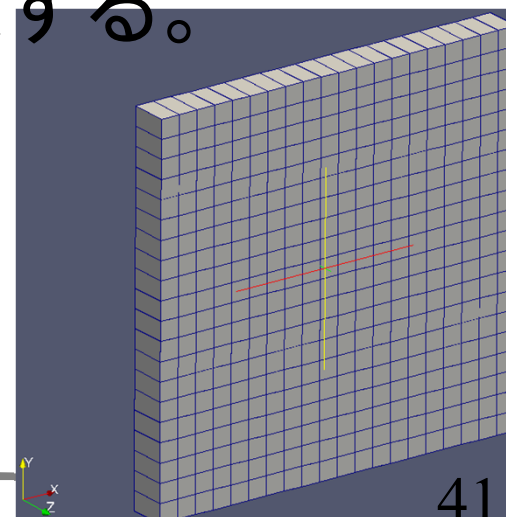
- 上部メニューで、「Surface」から「Surface Edges」or「Wireframe」に変更する。



メッシュの確認



- 左のような格子が表示されることを確認する。
- クリック&ドラッグでモデルを動かし、z方向の格子も確認する。



確認後、paraviewを終了する。

計算 Solving

ユーザマニュアル2.1.3節 p.26

計算を実行する

【作業：端末】

- ケース「cavity」ディレクトリにいることを確認するため、下記コマンドを実行する。

pwd

- /home/user/OpenFOAM/user-2.2.2/run/tutorials/incompressible/icoFoam/cavityと表示されればよい。違う場所にいるときは、下記コマンドを実行する。

```
cd $FOAM_RUN/tutorials/incompressible/icoFoam/cavity
```

- 下記のコマンドを実行する

icoFoam

- 計算のレポートが端末に表示される。ケースディレクトリに結果が出力される。(0.1から0.5)

ポスト処理 Post-processing

目 p.27

ユーザマニュアル2.1.4節 p.26

ParaView (オープンソースソフトウェア) を使って結果を可視化

OpenFOAMの結果を可視化するコマンドは
paraFoam

このコマンドは、OpenFOAMのケースディレクトリに「ケース名.OpenFOAM」というファイルを作り、ParaViewを起動する。

結果の可視化

目 p.27

ユーザマニュアル2.1.4節 p.26

ポスト処理ソフトParaViewを使って、メッシュを確認する

【作業：端末】

- ケース「cavity」ディレクトリにいることを確認するため、下記コマンドを実行する。

pwd

- /home/user/OpenFOAM/user-2.2.2/run/tutorials/incompressible/icoFoam/cavityと表示されればよい。違う場所にいるときは、下記コマンドを実行する。

```
cd $FOAM_RUN/tutorials/incompressible/icoFoam/cavity
```

- 下記のコマンドを実行する

paraFoam

- ParaViewが起動する

圧力の可視化

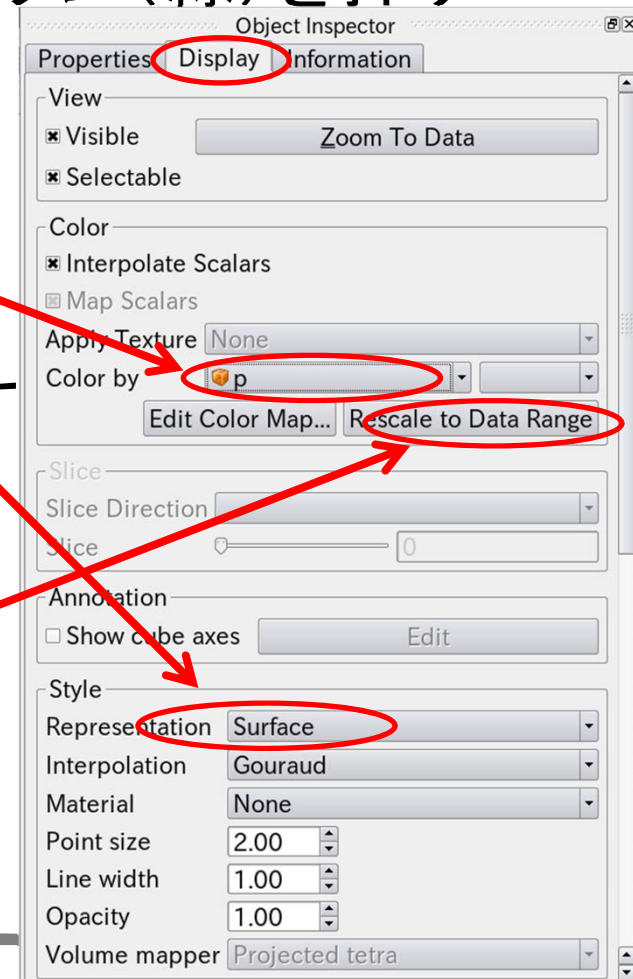
ユーザマニュアル2.1.4.1節 p.28

【作業: ParaView】

- ParaViewが起動したら、Applyボタン(緑)を押す
- 「Object Inspector」ウィンドウで、Color by を p Style を Surface にする。
- 上部メニューの再生ボタンを押す



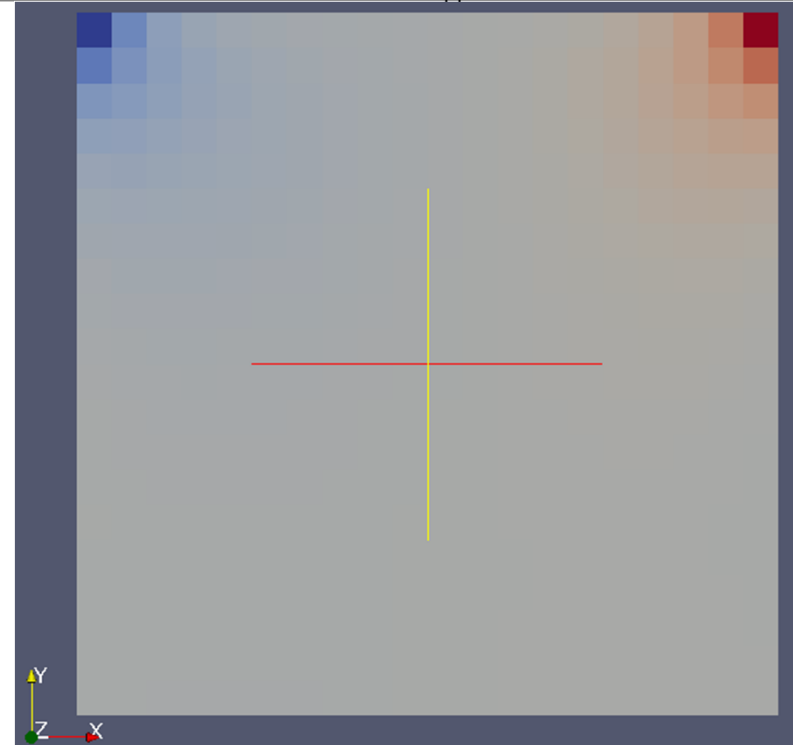
- 「Object Inspector」ウィンドウで、Rescale to Data Rangeを押す



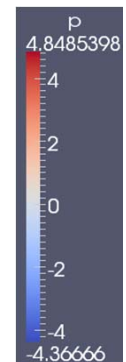
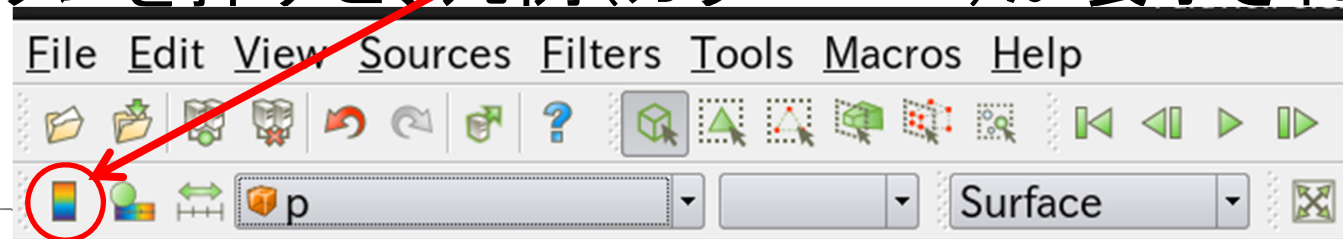
圧力の可視化

ユーザマニュアル2.1.4.1節 p.28

- 圧力分布が表示される。
- 「Edit Color Map」
→「Choose Preset」
→「上から2番目」で
青から赤の虹色表示

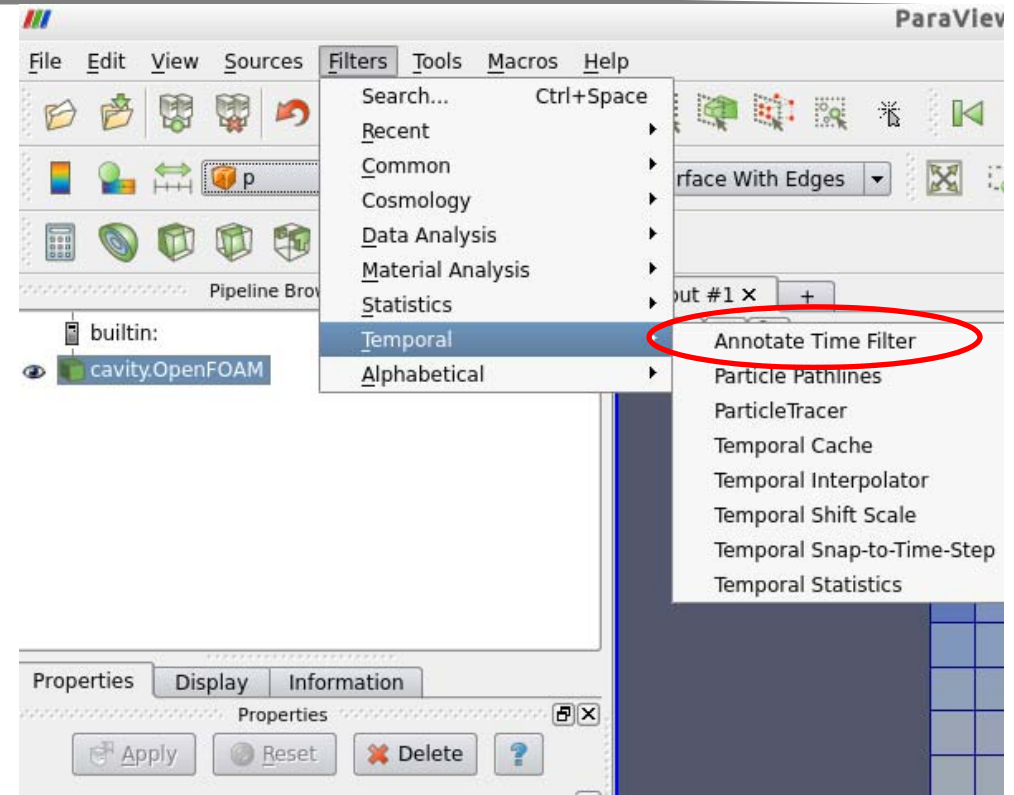


- 上部メニューの「Toggle Color Legend Visibility」ボタンを押すと、凡例(カラーバー)が表示される。



時刻の表示: Annotate Time フィルター

- Pipeline Browserで cavity.OpenFOAMをハイライト
- Filters — Temporal と進み、Annotate Time Filterをクリック
- Applyボタンを押す
- 表示している結果の時刻が書かれる。クリック&ドラッグ可能。

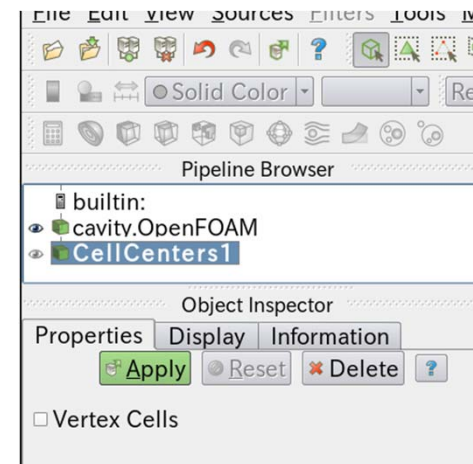
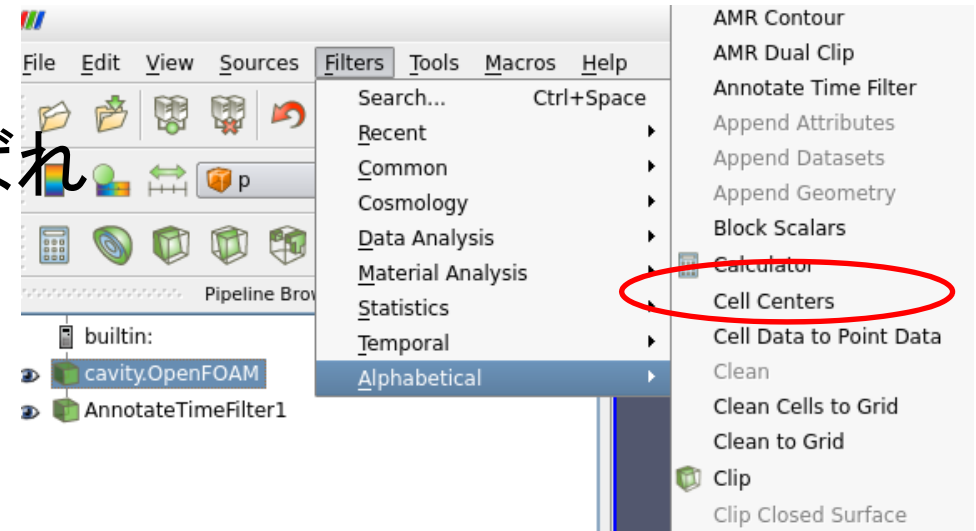


速度ベクトル図 Glyph (1)

ユーザマニュアル2.1.4.2節 p.28

セル中心での速度ベクトル

- Pipeline Browserで cavity.OpenFOAMが選ばれていることを確認
- Filtersメニューから、Alphabeticalと進み、CellCentersをクリックする。
- Object Inspectorで、Propertiesタブを選び、Applyボタンを押す

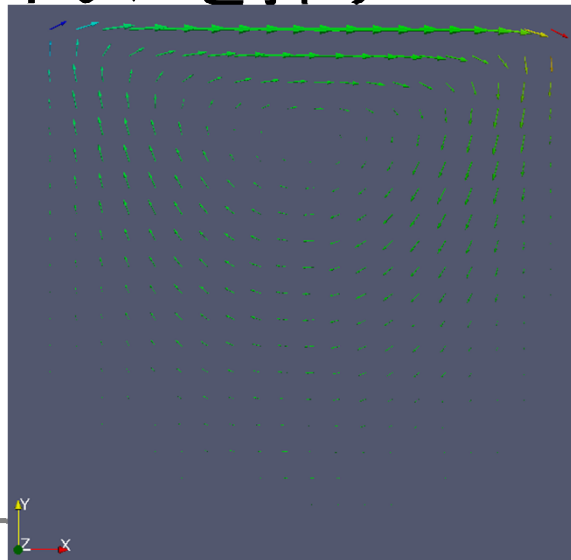
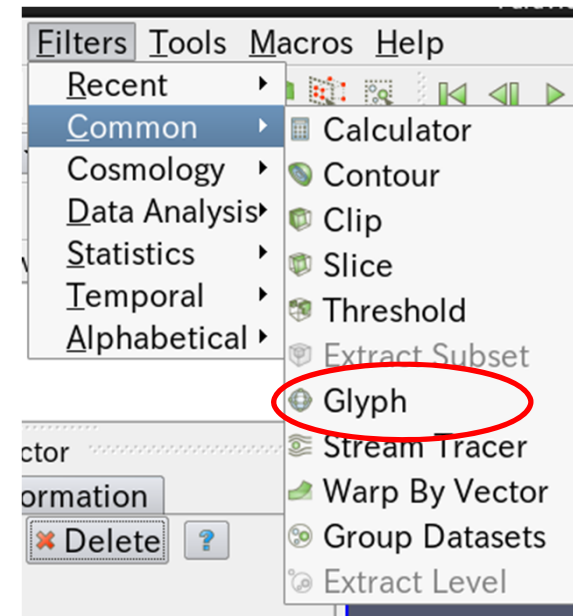
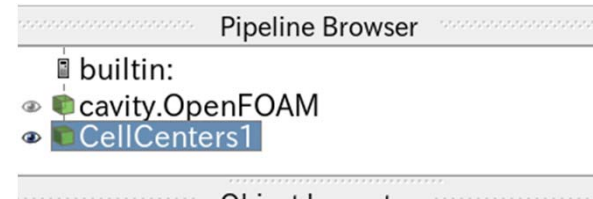


続く。。。

速度ベクトル図 Glyph (2)

ユーザマニュアル2.1.4.2節 p.28

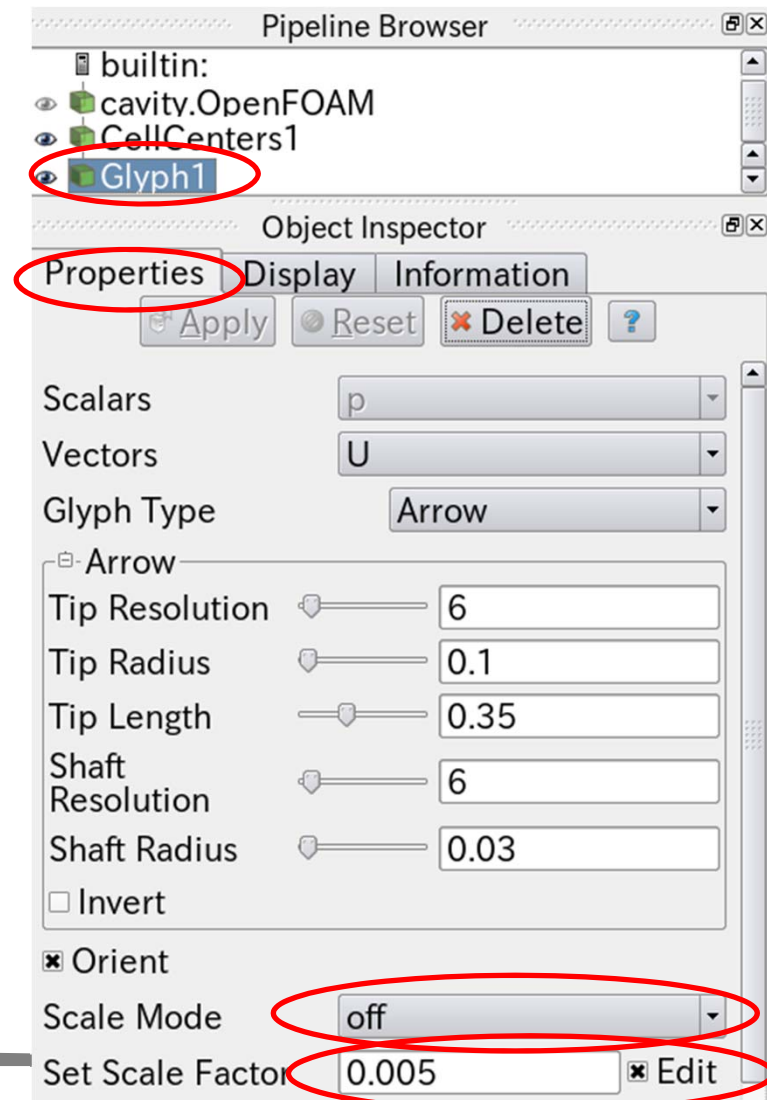
- Pipeline Browserで CellCentersが選ばれていることを確認
- Filtersメニューから、Commonと進み、Glyphをクリックする。
- Applyボタンを押す



速度ベクトル図 Glyph (3)

ユーザマニュアル2.1.4.2節 p.28

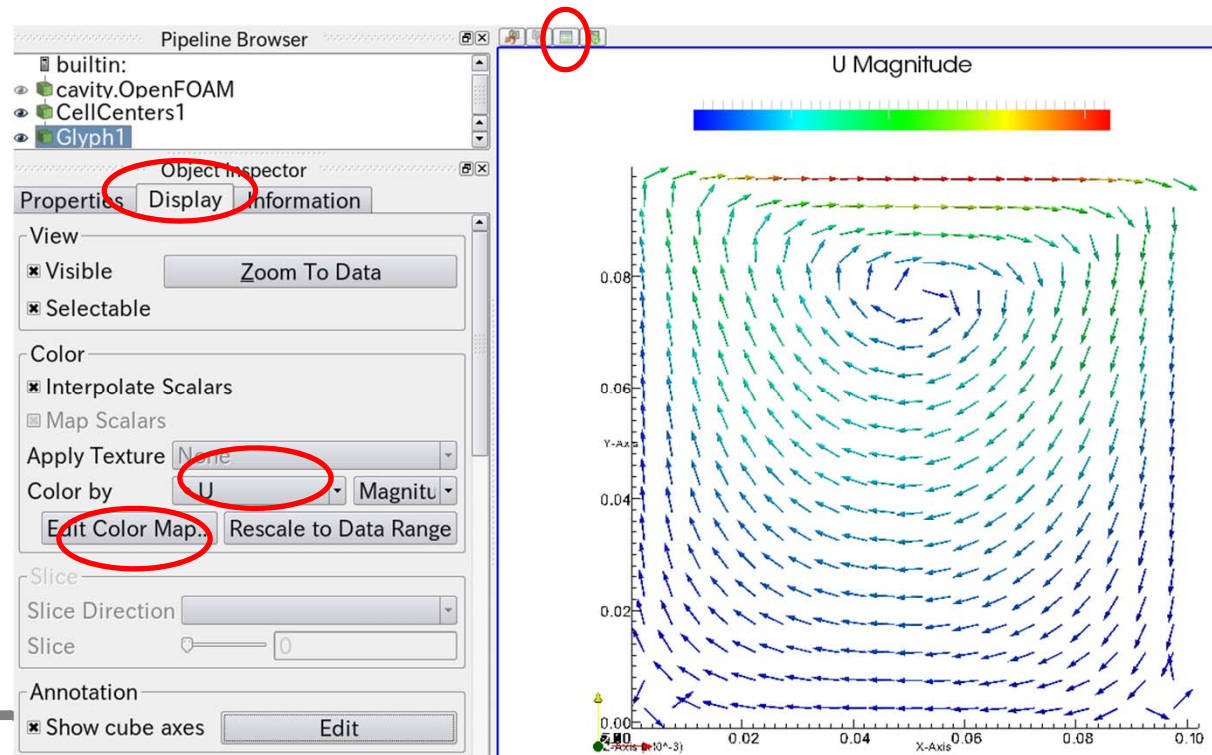
- 標準では、速度の大きさに合わせて、ベクトルの長さが決まる。
- 速度の遅い部分を見やすくするため、ベクトルの長さを統一する。
- PipelineBrowserでGlyph選択、Propertiesタブをクリックし、
Scale Mode を off に、
SetScaleFactor のEditを
チェックして、数字を0.005に。



速度ベクトル図 Glyph (4)

ユーザマニュアル2.1.4.2節 p.28

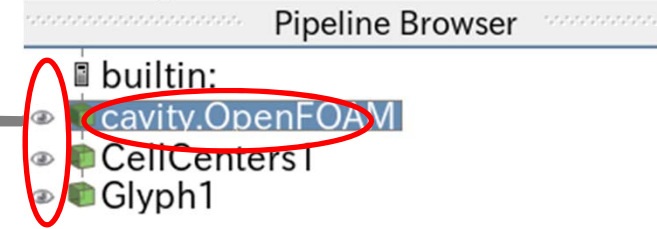
- PipelineBrowserでGlyph選択、**Display**タブをクリックし、Color by を U に、 EditColorMap で BlueToRed
- 背景色はEditViewOptionから変更可能。



流線 Streamline

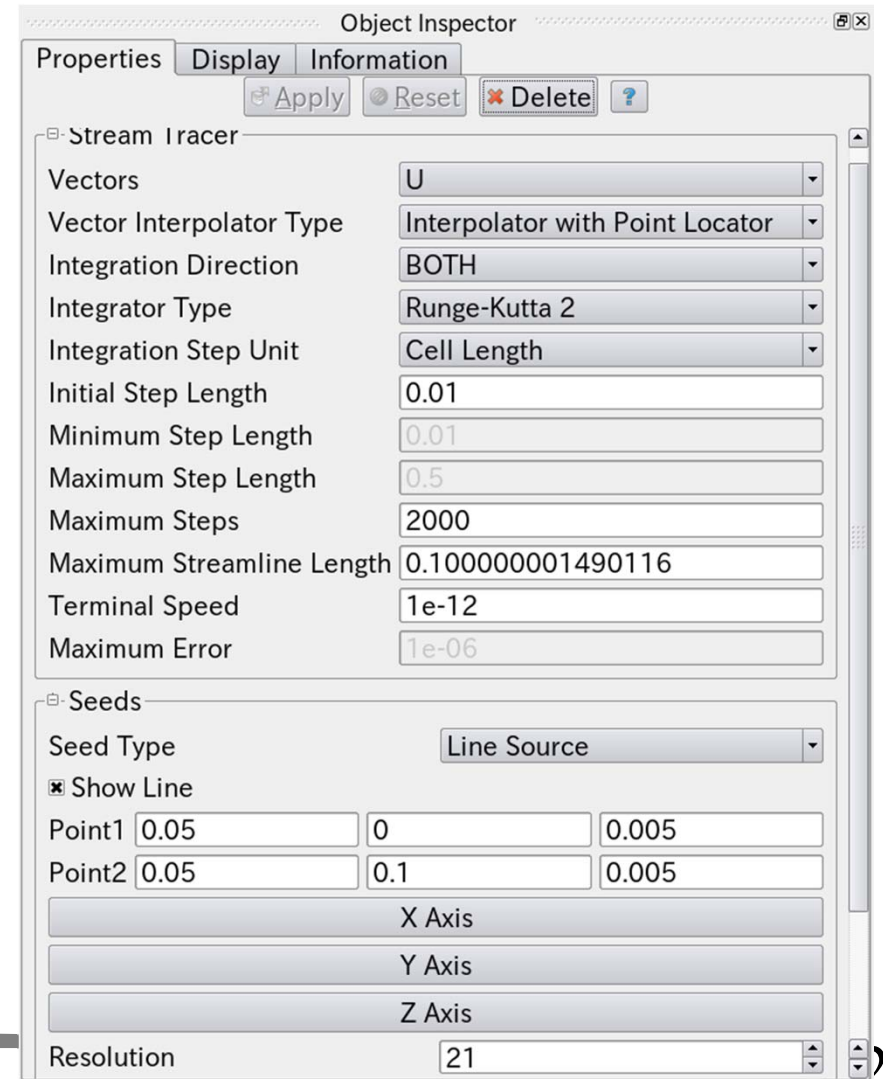
目 p.29

ユーザマニュアル2.1.4.3節 p.30



流線

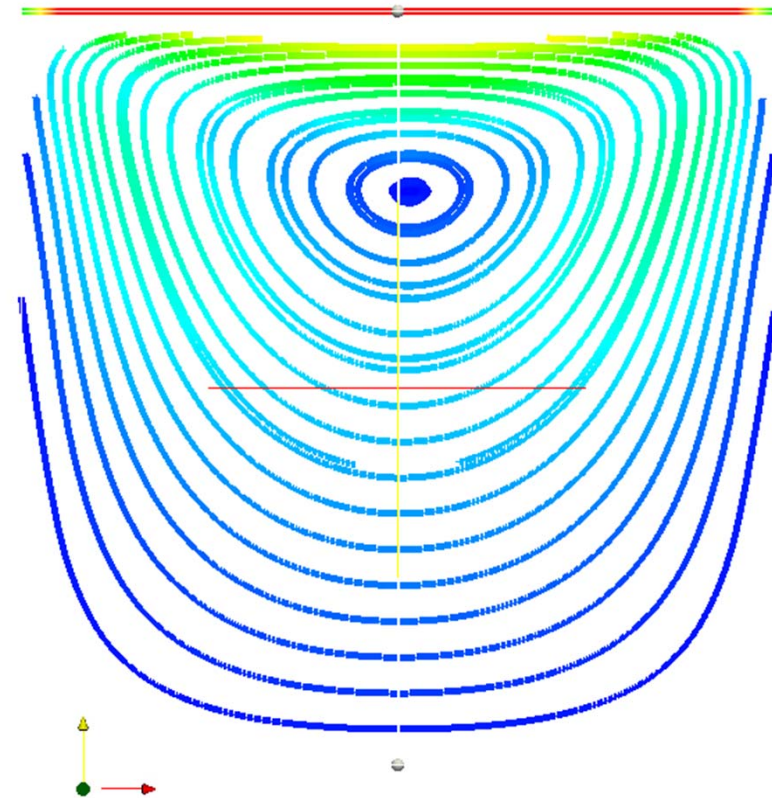
- Pipeline Browserで cavity.OpenFOAMが選ばれていることを確認
- すべてのfilterの表示をoffにするため、目玉をクリックして消す
- Filtersメニューから、Commonと進み、StreamTracerをクリックする。
- 右のようにパネルを設定
- Applyボタンを押す



流線 Streamline

流線

- StreamTracerのDisplayタブで、StyleのLine widthを大きくすると、線が太くなる。



可視化結果の保存

- 画像の保存
 - メニュー: File — Save screenshot
- 動画の保存
 - メニュー: File — Save Animation
- 作業状態の保存
 - メニュー: File — Save State

ケースのコピー

ユーザマニュアル2.1.5.1節 p.30

【作業：ファイルマネージャ】

現在のケースディレクトリ

(/home/user/OpenFOAM/user-

2.2.2/run/tutorials/incompressible/icoFoam/cavity)をコピーして、貼り付ける。

名前を cavityFine とする。

/cavityFine/constant/polyMesh/blockMeshDictをダブルクリックし、ファイルを開く。

メッシュの細分化

目 p.31

ユーザマニュアル2.1.5.2節 p.32

- /cavityFine/constant/polyMesh/blockMeshDictをダブルクリックし、ファイルを開く。
- Blocks部分を下記のように変更して保存する。

```
blocks  
(  
  hex (0 1 2 3 4 5 6 7) (40 40 1) simpleGrading (1 1 1)  
);
```


計算パラメータ修正

目 p.32

ユーザマニュアル2.1.5.4節 p.33

- /cavityFine/constant/system/controlDictをダブルクリックし、ファイルを開く。
- 時間刻みを 0.005 から 0.0025にする。
 - セルが半分の大きさになったため、クーラン数を1以下にするには、時間も半分にする必要がある。
- writeInterval を 20 から 40 に変更する。
 - 先ほどと同じ間隔(0.1秒毎)でデータを書き出すため。
($0.0025 \times 40 = 0.1$)
 - あるいは、writeControl を runTime とし、writeInterval を 0.1 としてもよい。(書き出す時刻を直接指定する。)

計算(メッシュ細分化)

計算を実行する

ユーザマニュアル2.1.5.5節 p.33

【作業】

- ケース「cavityFine」ディレクトリを右クリック, 端末で開く。
- 下記のコマンドを実行する(メッシュの生成, バックグラウンド実行とログのファイルへの書き出し)

```
blockMesh
```

```
icoFoam > log &
```

- 計算のレポートが端末に表示されるかわりに, ケースディレクトリにlogという名のファイルが生成される。
- ケースディレクトリに結果が出力される。(0.1から0.5)