

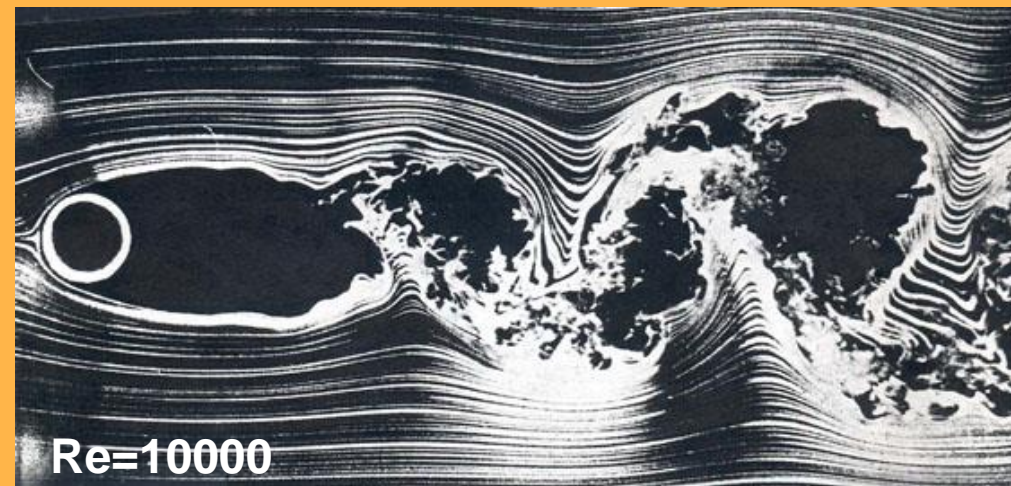
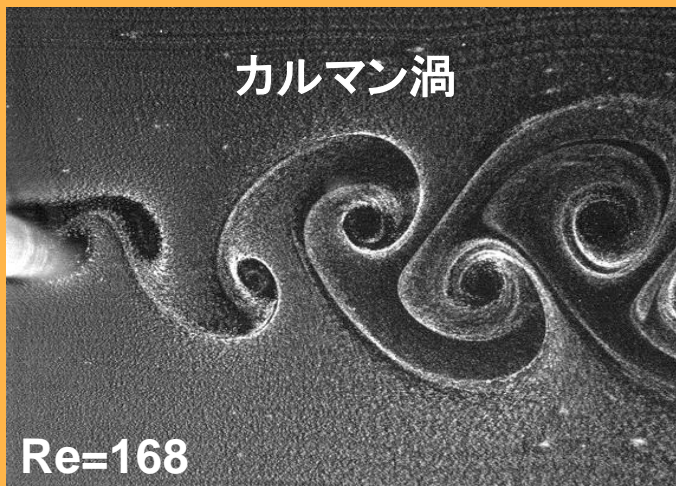
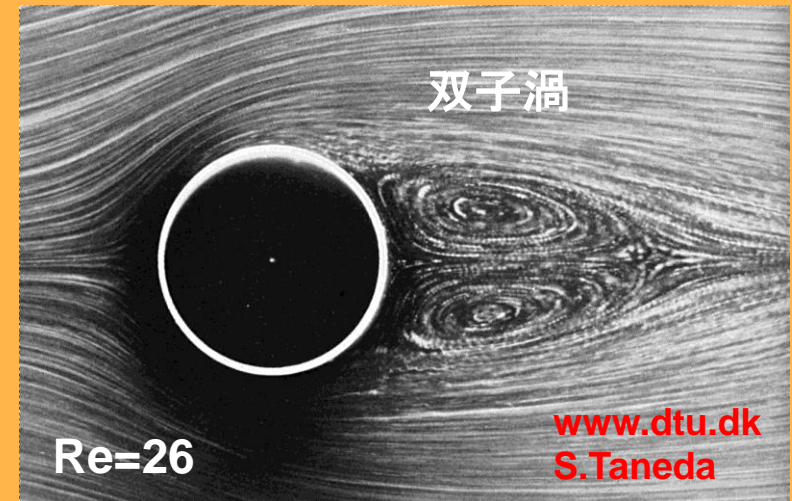
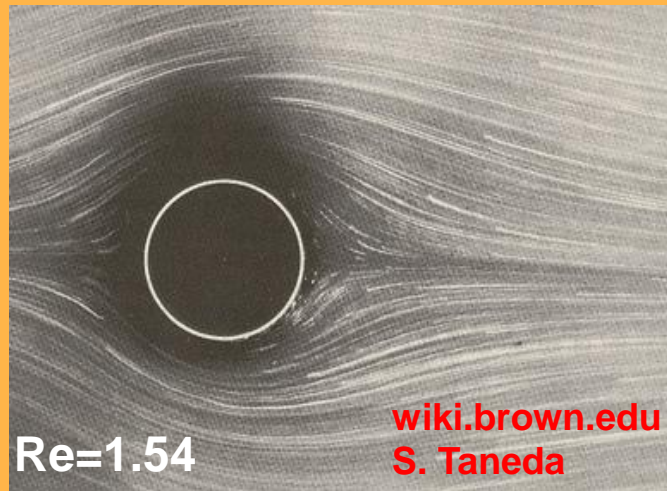
OpenFOAM演習

pisoFoamによる 2次元円柱周りの流れの解析 (修正版)

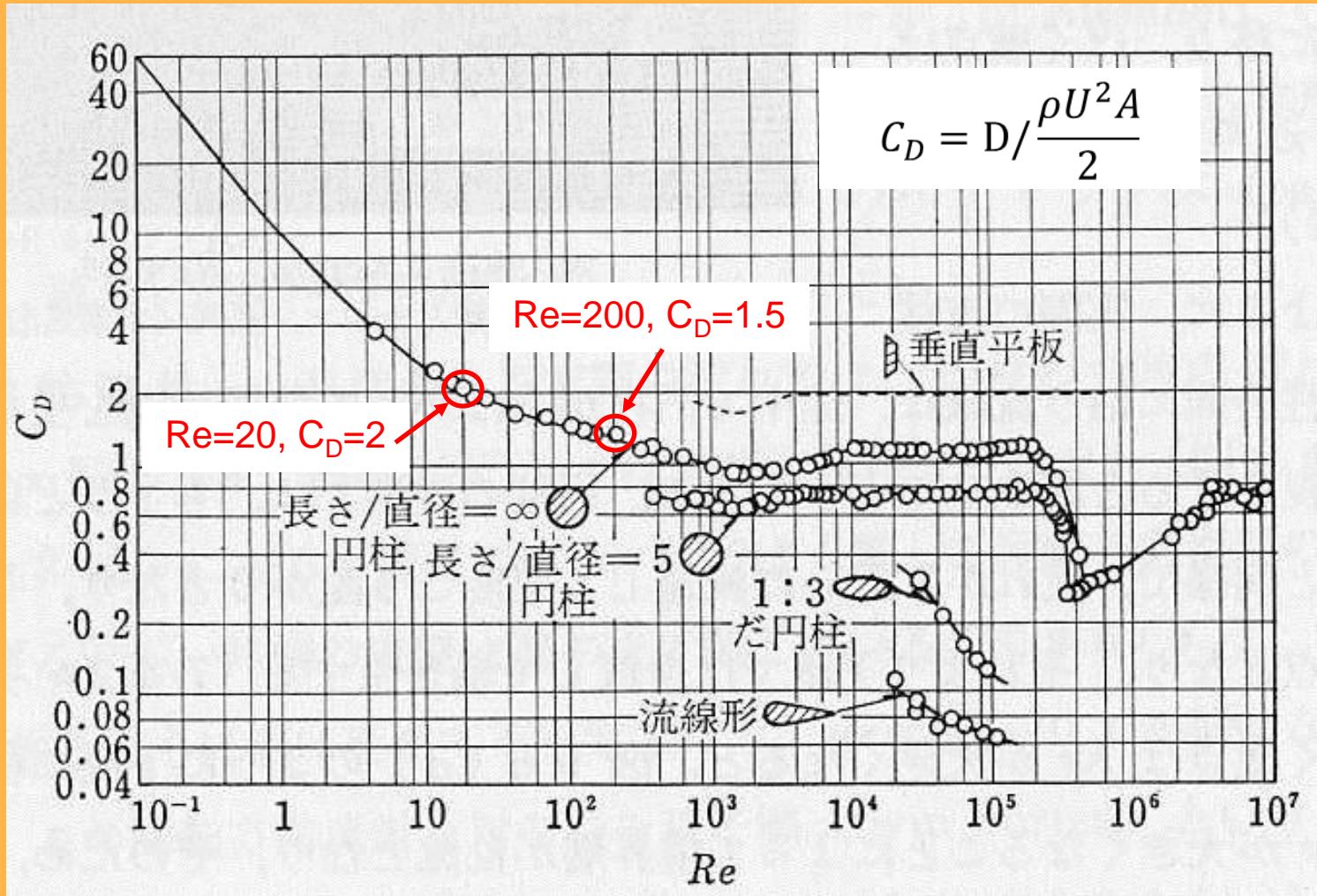
田村

第24回OpenCAE勉強会(岐阜)

Reによる流れの変化



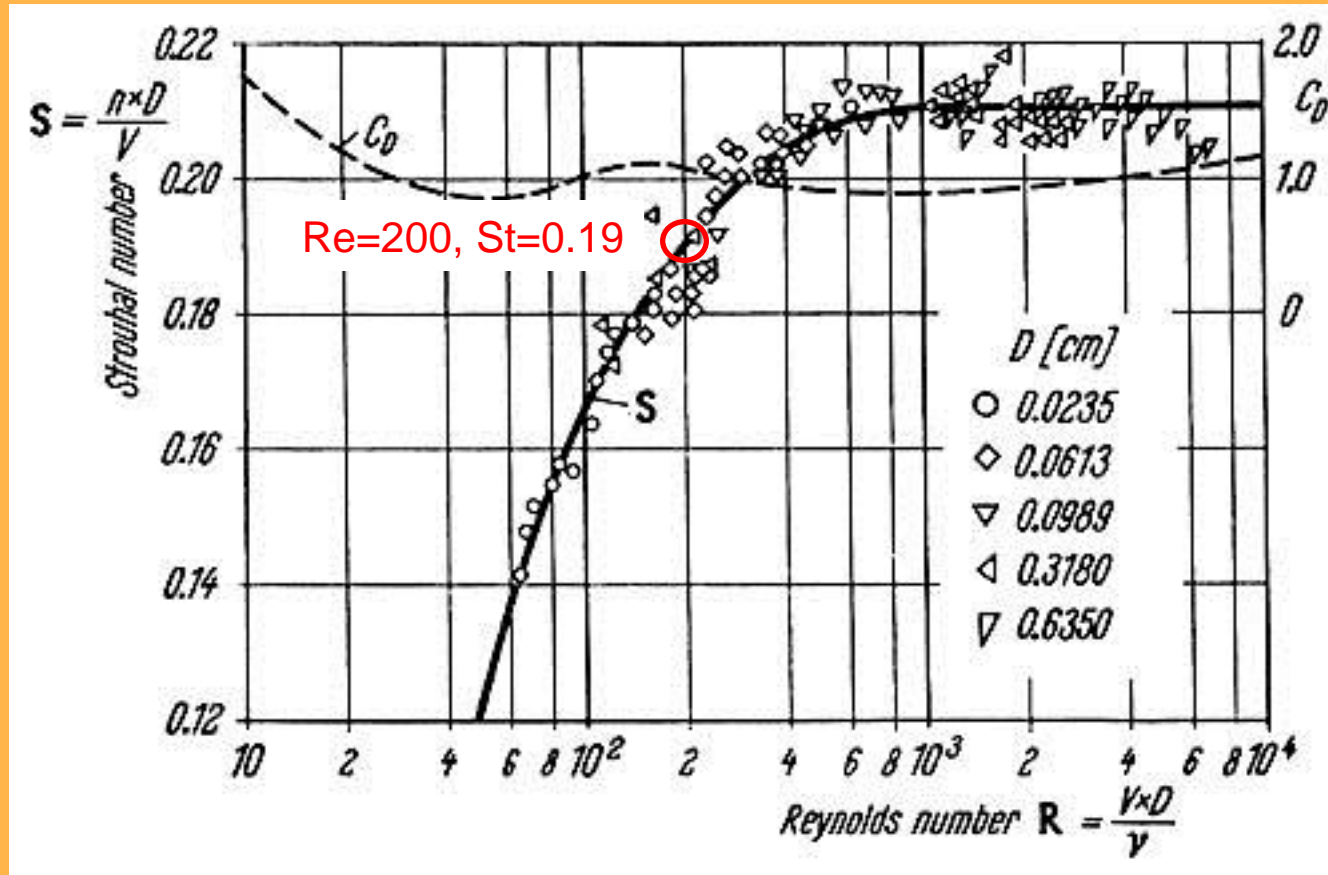
抗力係数



流体の力学: 中山泰喜著 養賢堂 (1998) P138

ストローハル数

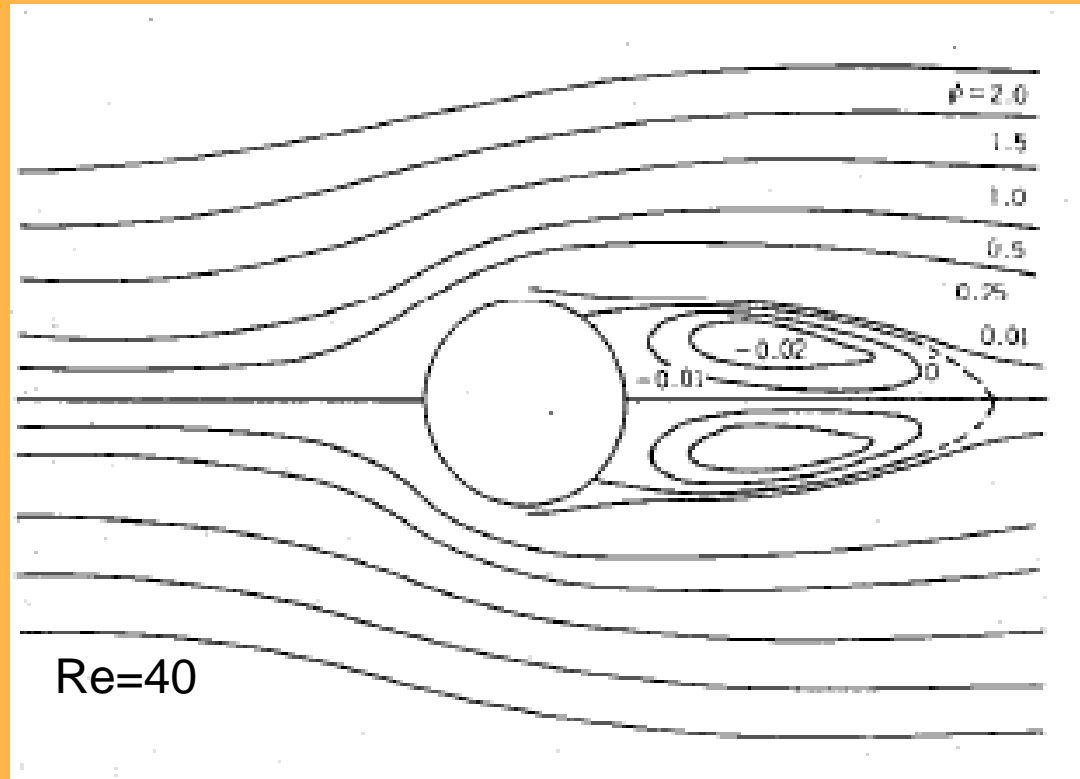
<http://www.dept.aoe.vt.edu/~jschetz/fluidnature/unit09/unit9a.html>



The Strouhal number in terms of the Reynolds number of the flow past a circular cylinder as measured by A. Roshko

NS方程式の数値計算の草分

: 川口光年さんの計算(1953)

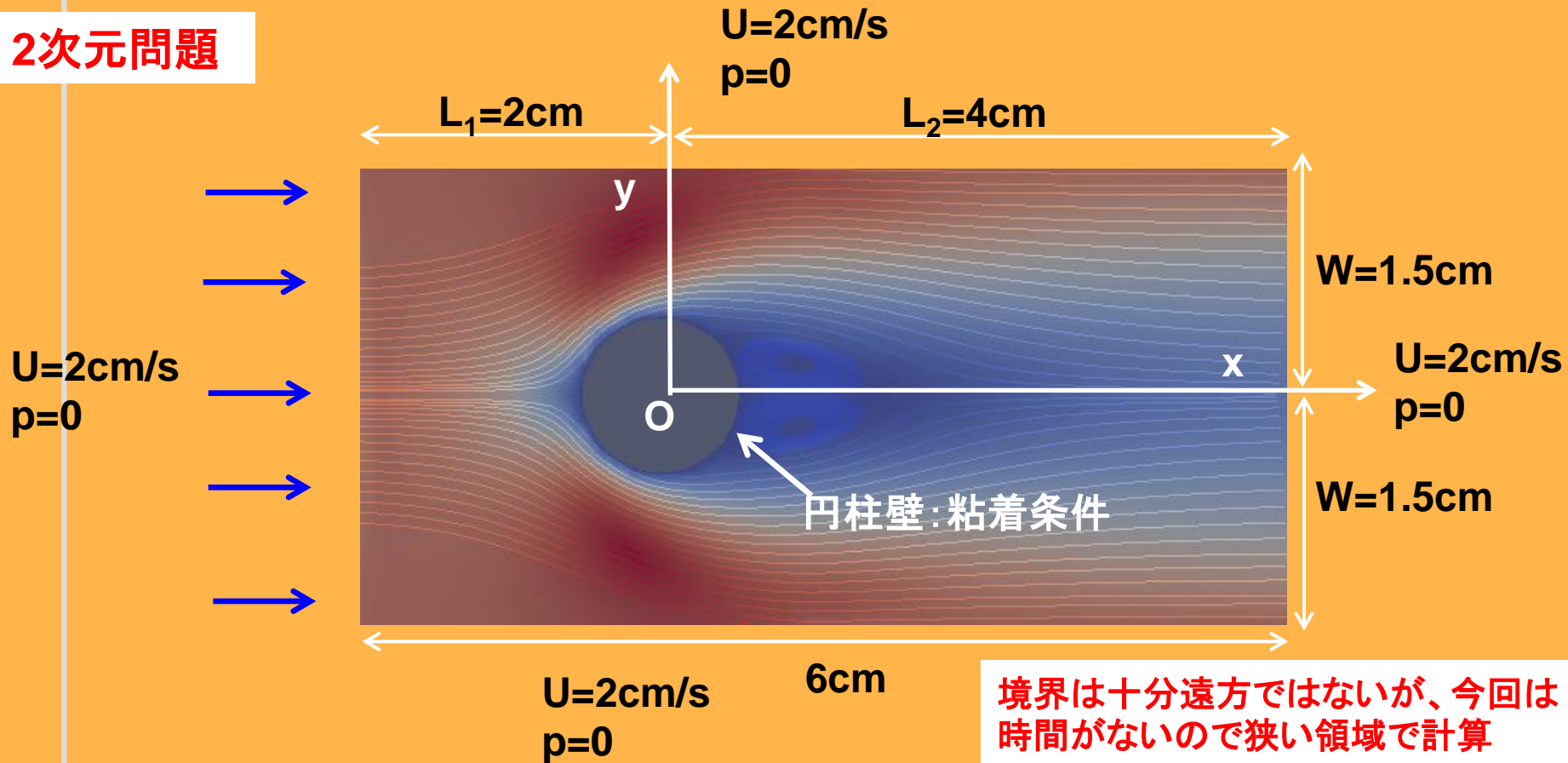


週20時間、1年半タイガー計算機で計算。

Kawaguti, M.: J. Phys. Soc. Jpn. 8 (1953), 747.

例題(円柱の C_D 値を求める)

2次元問題



円柱直径 $D=1\text{cm}$ 流体は水⇒非圧縮Newton流体
動粘性係数 $\nu=1 \times 10^{-5}\text{m}^2/\text{s}$
レイノルズ数 $Re=UD/\nu=20 \Rightarrow$ 層流

支配方程式

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

$$\frac{\partial v}{\partial t} + \frac{\partial v^2}{\partial x} + \frac{\partial v^2}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)$$

OFのデフォルトで条件
設定はできない

(upstream) $u = 0.02, v = 0, p = 0$

(downstream) $u = 0.02, v = 0, p = 0$

(up & down) $u = 0.02, v = 0, p = 0$

(cylinder wall) $u = 0, v = 0$ $\frac{\partial p}{\partial n} = \rho \nu \frac{\partial^2 v_n}{\partial n^2}$

pisoFoam

OpenFOAM ver.2.1.x tutorials
incompressible/pisoFoam

<前提>

- ・非定常問題
- ・非圧縮性層流
- ・Newton流体/非Newton流体
- ・非圧縮性乱流モデル(RAS, LES)

<pisoFoamの由来>

次の非定常解法を使う

PISO algorithm – Pressure Implicit with Split Operator

演習の流れ

円柱stlファイルの作成 (FreeCAD)



メッシュ作成 I (blockMesh)



メッシュ作成 II (snappyHexMesh)



境界条件、計算条件の設定



計算実行 (pisoFoam)



計算結果の表示 (paraFoam)

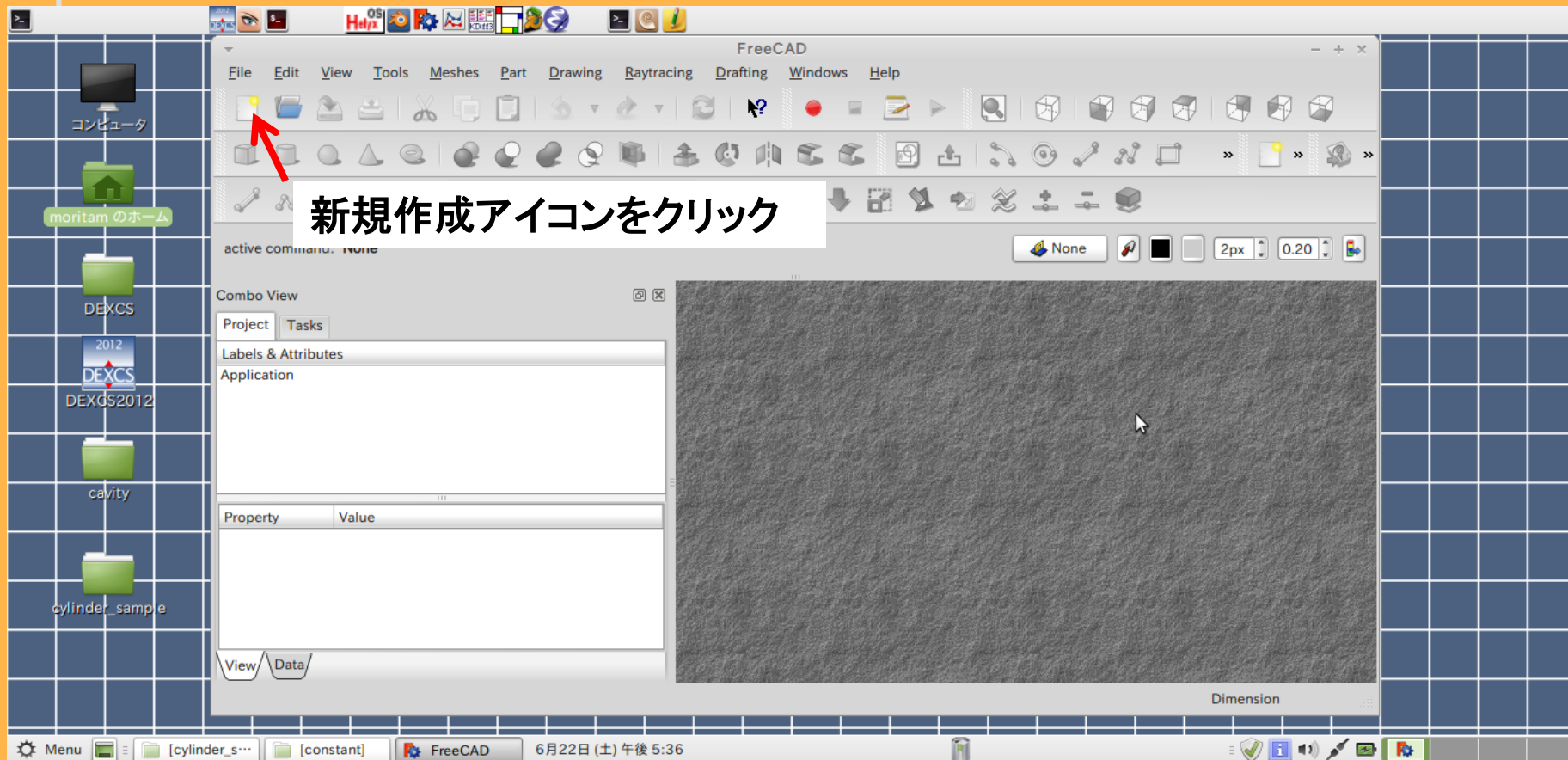
円柱のstlファイル作成

円柱作成1 (FreeCAD起動)



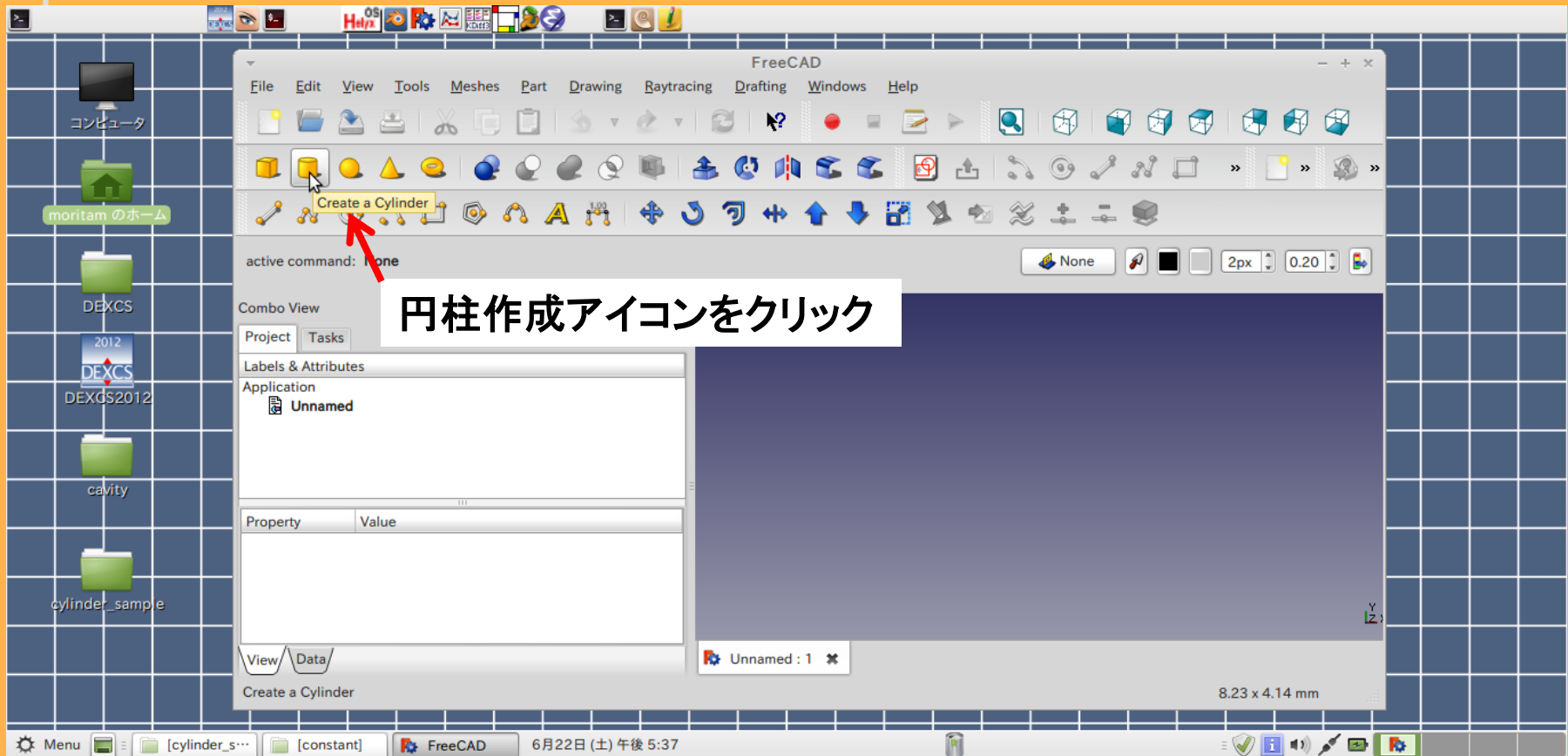
作業場所 : Desktop

円柱作成2(新規作成)

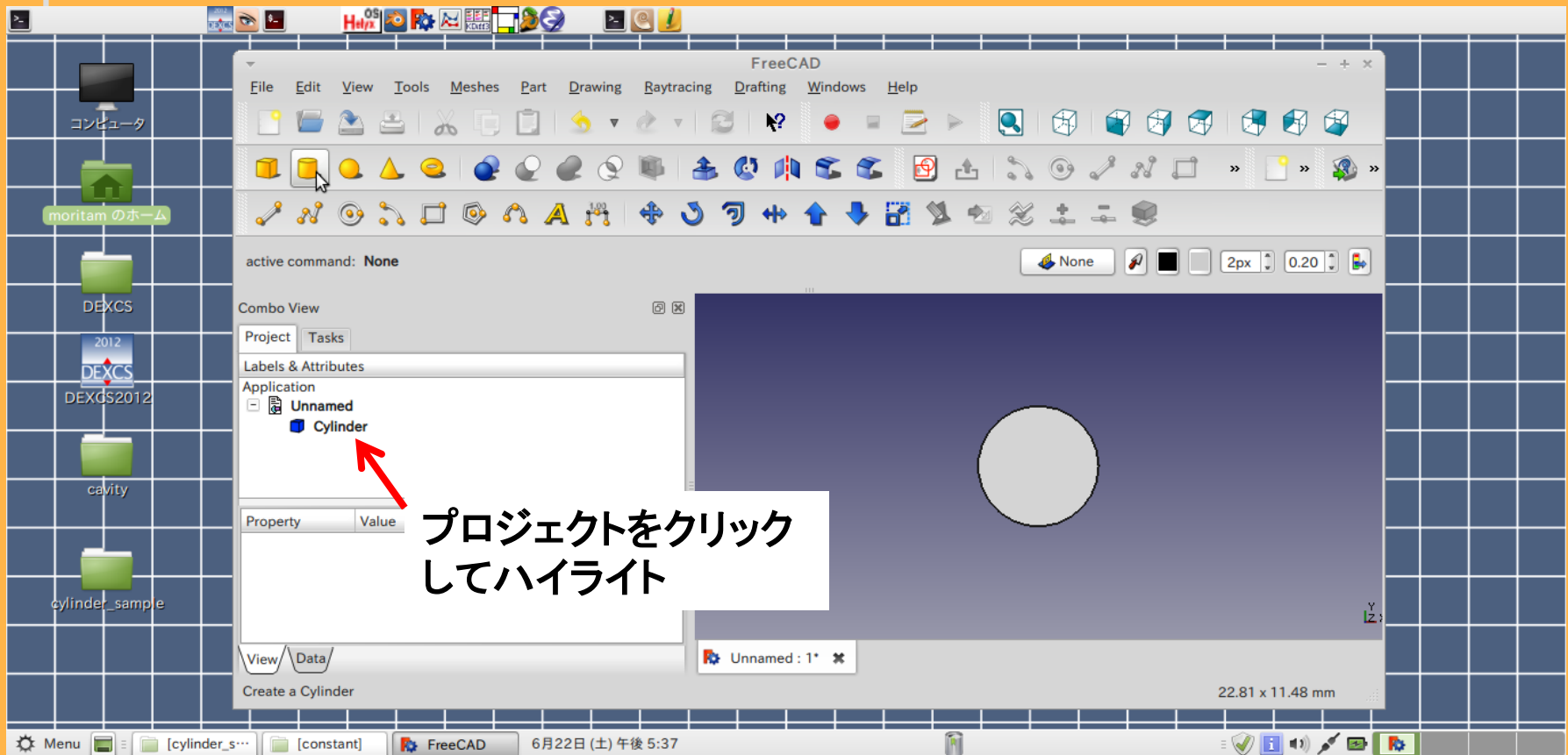


起動時の画面

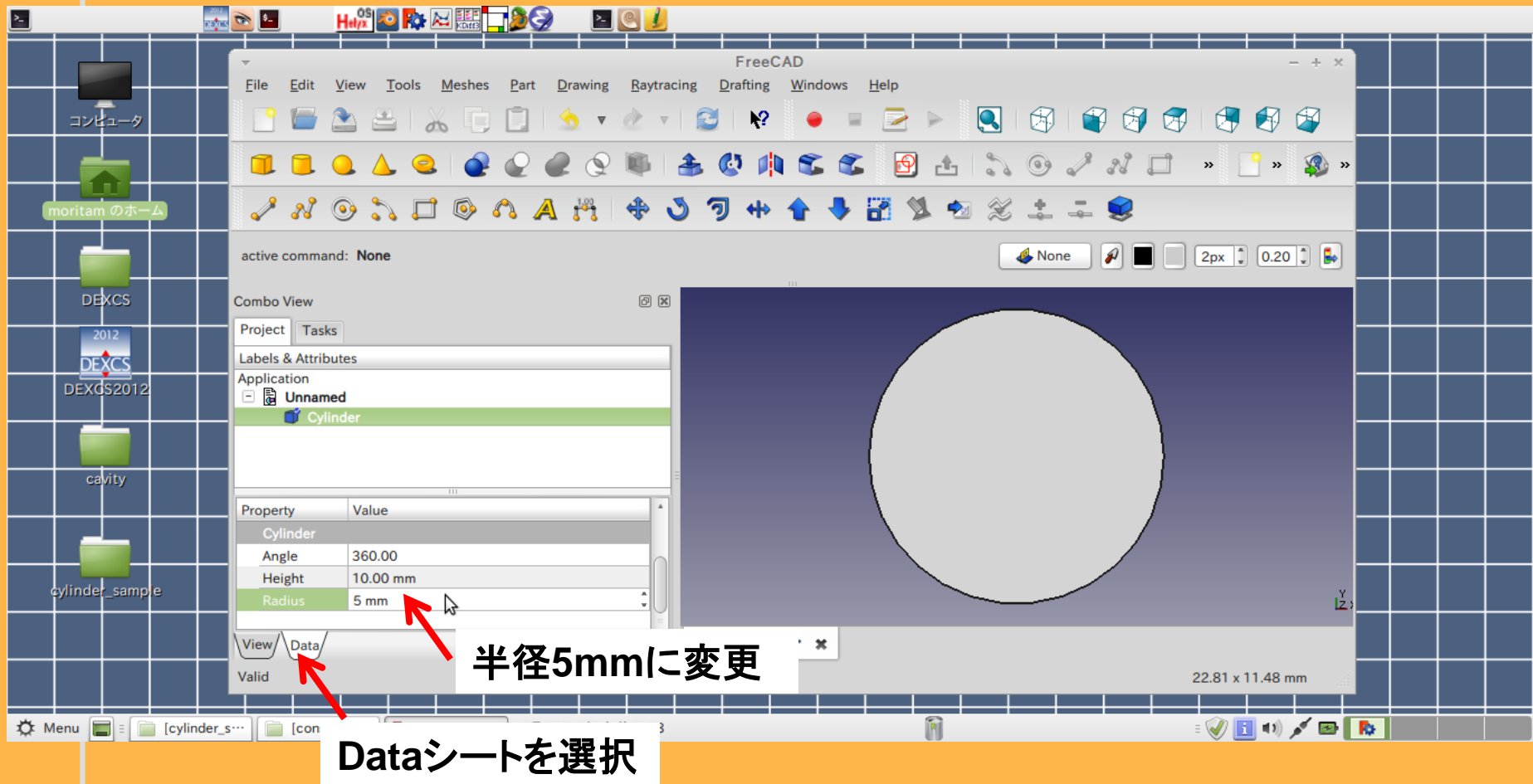
円柱作成3(円柱作成アイコン)



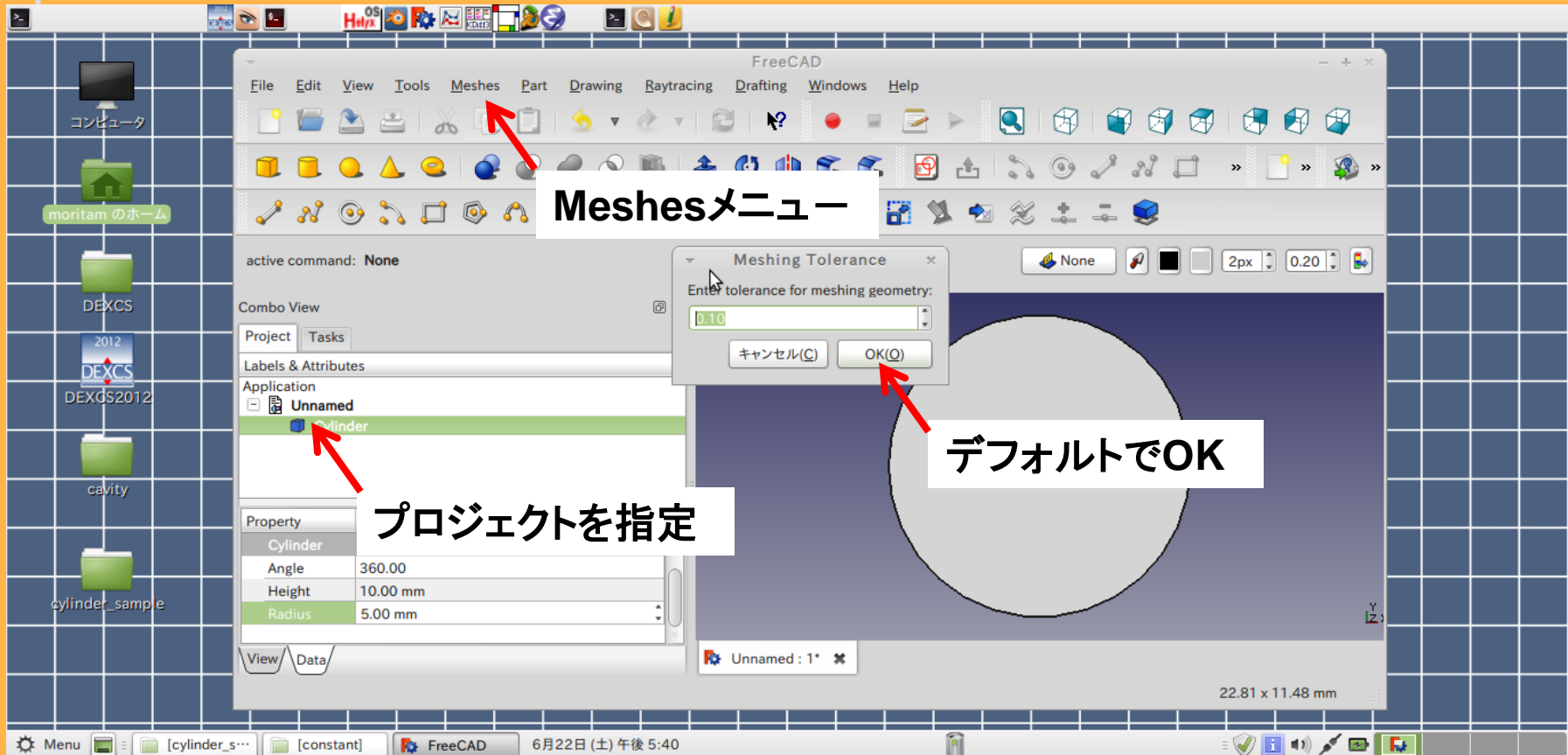
円柱作成4(プロジェクトの指定)



円柱作成5(寸法指定)



円柱作成6(メッシュ作成)

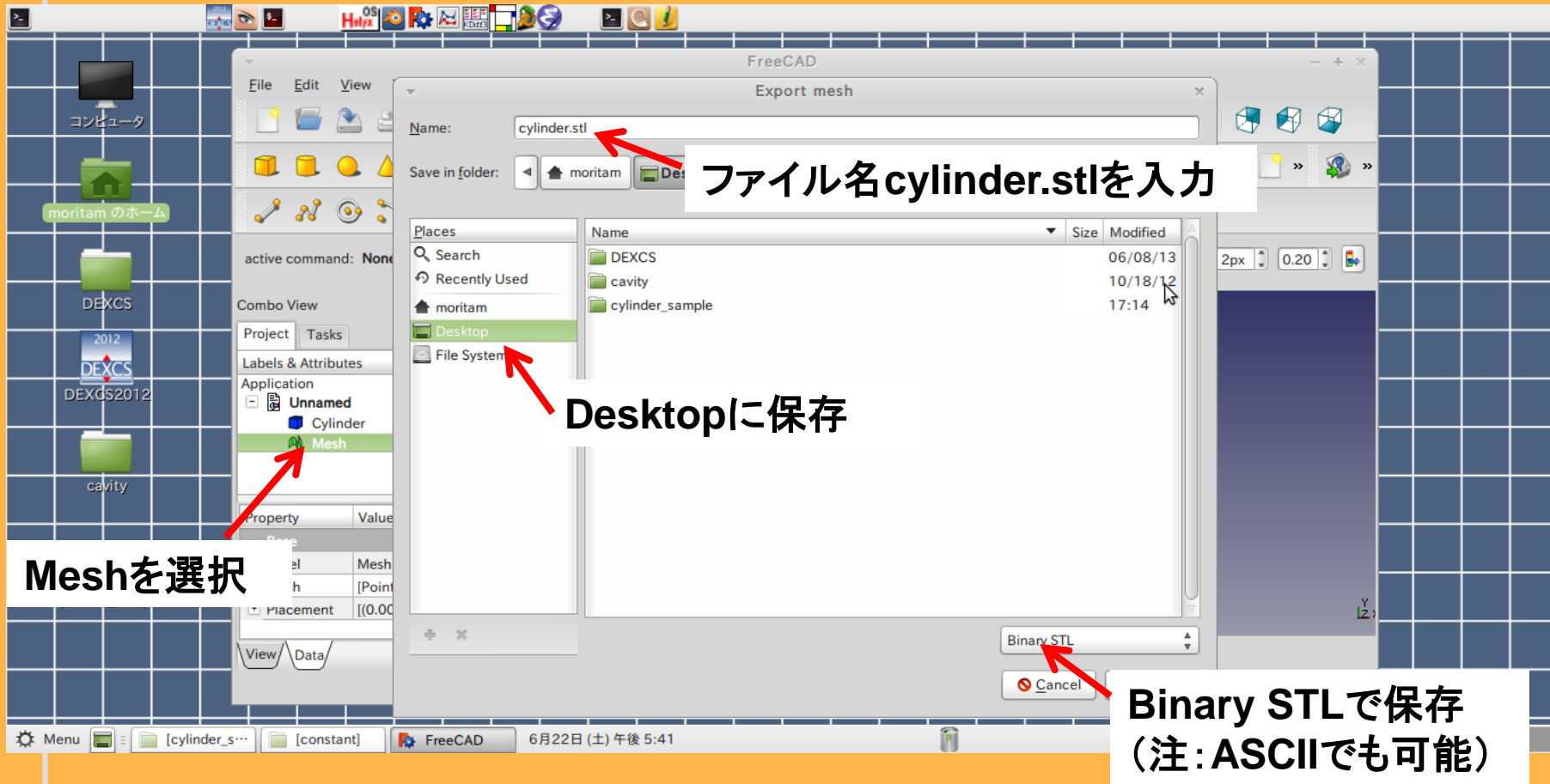


Meshesメニューの中でCreate mesh from geometryを選択
(注: Create mesh from shapeでも可能)

Meshing Toleranceメニューがでる。

メッシュサイズを制御するメニューだがここではデフォルトでOK

円柱作成7 (stlファイル保存)



Meshesメニューの中でExport Meshを選択
Export meshメニューがでる。

ファイル名 cylinder.stl を指定しBinary STLでDesktopに保存

円柱作成8(単位変換)



- ✓ FreeCADで半径10mmの円柱stlを作ったが単位系はOpenFoamで認識しない
- ✓ OpenFoamではmモジュールを使うので1/1000に変換が必要
- ✓ 端末アイコンをクリック(3つあるが、OpenFOAMコマンドが使えるのは真ん中)
- ✓ Desktopに移動し、以下のコマンドを使って単位変換

\$cd Desktop

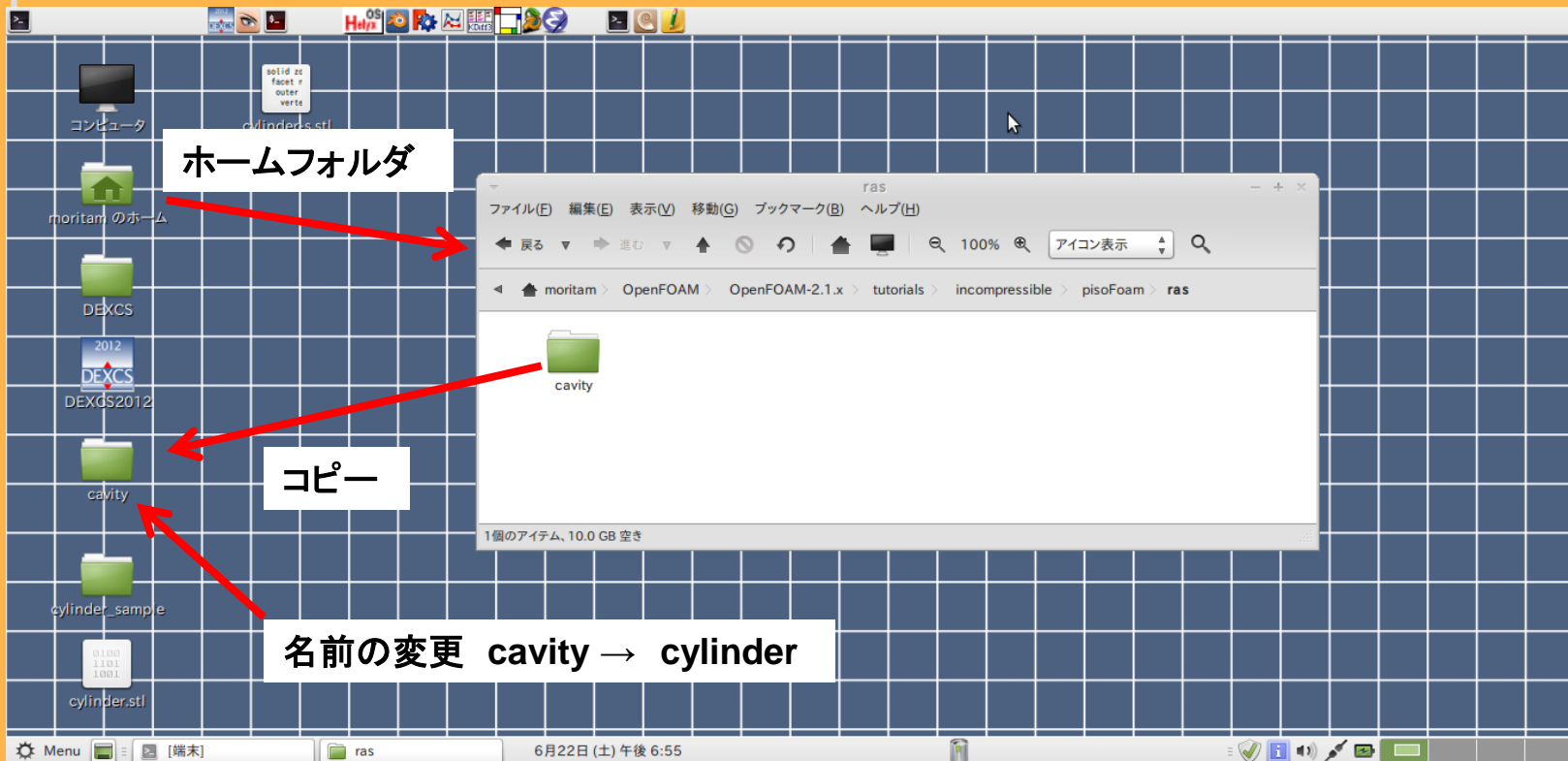
\$surfaceTransformPoints cylinder.stl cylinder-s.stl -scale '(0.001 0.001 0.001)'

2013/8/10

ベースケースの取得

pisofFoam/ras/cavity

ベースtutorial



ベースのtutorialのディクショナリをDesktopにコピー
ホーム/OpenFOAM/OpenFOAM-2.1.x/tutorials
/incompressible/pisoFoam/ras/cavity

ディクショナリ構造の概要



0

(時刻ディクショナリ)

U, p など物理量の $t=0$ の時の値、境界条件



constant

(constantディクショナリ)

polyMesh/blockMeshDict: 領域、メッシュ指定

transportProperties: 輸送物性値の指定

turbulenceProperties: 乱流モデル(RAS, LES)指定

RASProperties: レイノルズ平均モデルの指定



system

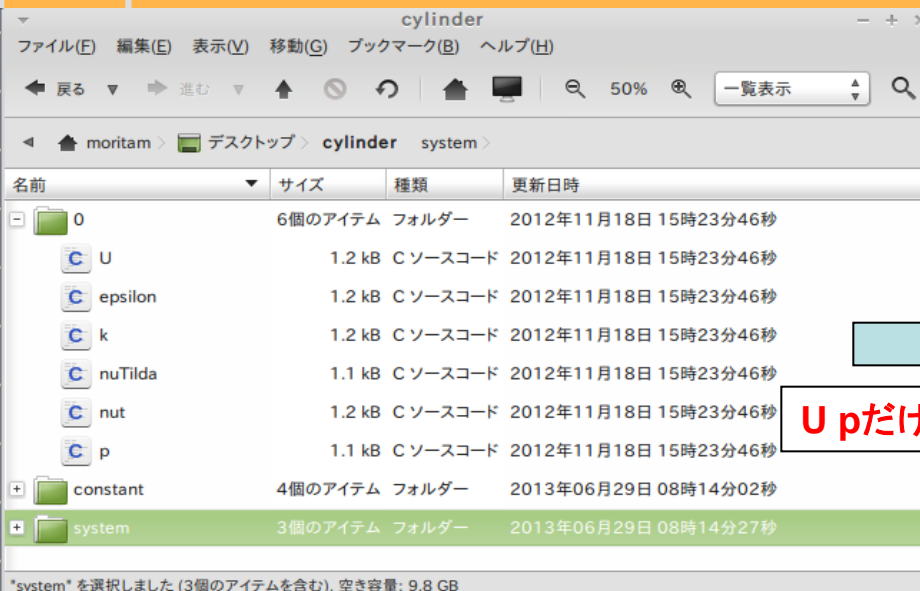
(systemディクショナリ)

controlDict: ソルバー、開始時間、終了時間、
タイムステップ等の指定

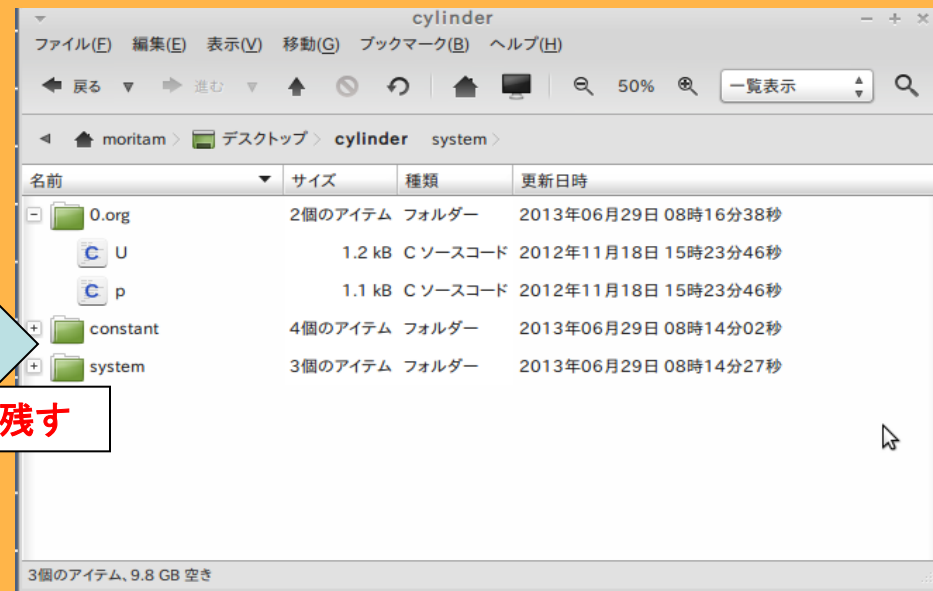
fvScheme: 有限体積法の離散化方法の指定

fvSolution: 有限体積法の方程式解法と収束条件の指定

0ディクショナリ



U pだけを残す

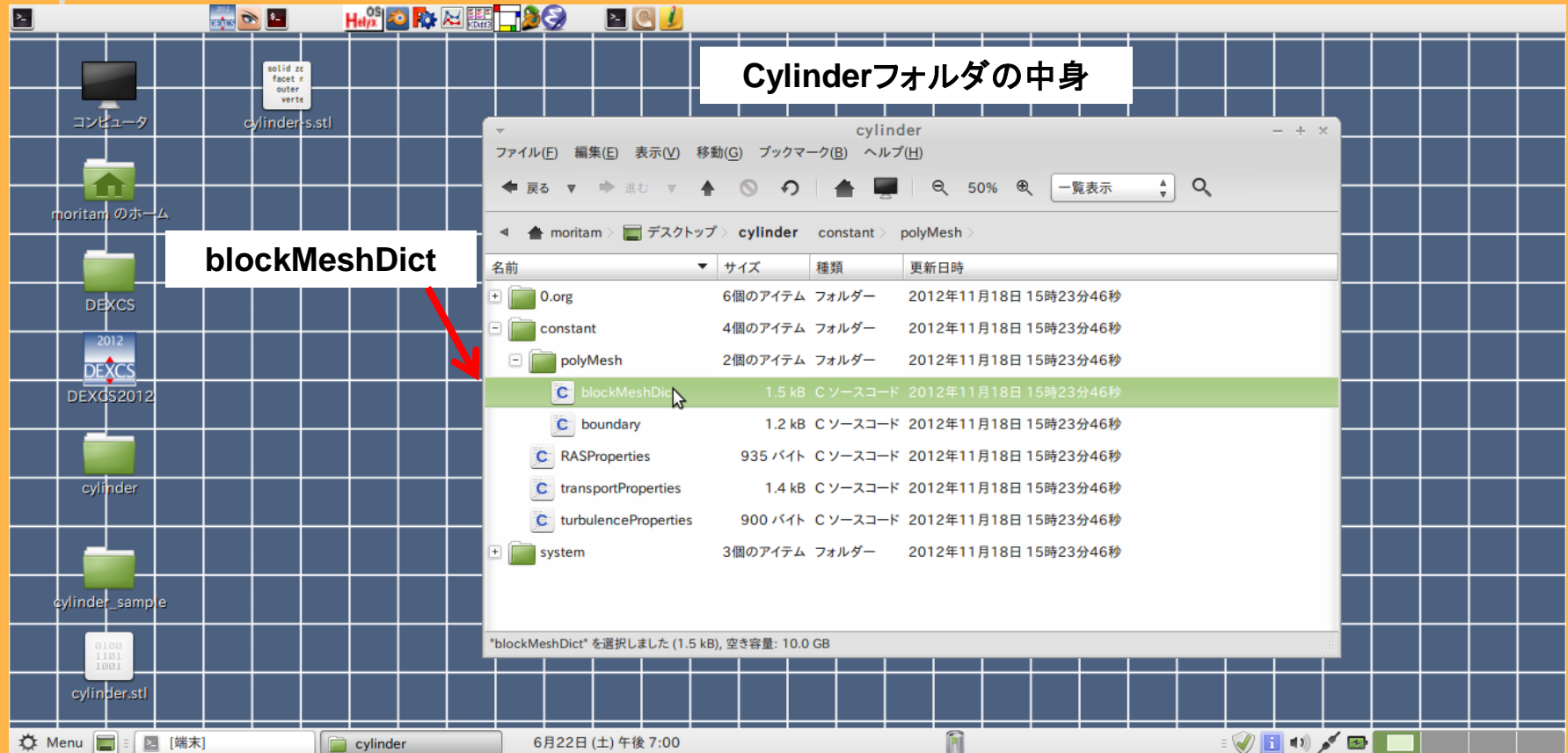


- ✓ 0ディクショナリには初期値および境界条件が格納
- ✓ U(速度)、epsilon (乱流エネルギー散逸率)、k(乱流エネルギー)、nuTilda(渦粘性: Spalart-Allmarasモデルで使用)、nut(渦粘性)、p(圧力)
- ✓ U、p しかここでは使わないので他は削除(置いてあっても問題ない)
- ✓ 名称を 0 ⇒ 0.org に変更(0のままだとsnappyHexMeshでエラー)

BlockMesh作成

基本領域の6面体の設定とメッシュ分割

blockMeshDictの格納場所



- ✓ constant/polyMeshの中に基本の6面体メッシュ作成を制御するblockMeshDictがある。

基本領域設定 (blockMeshDict)

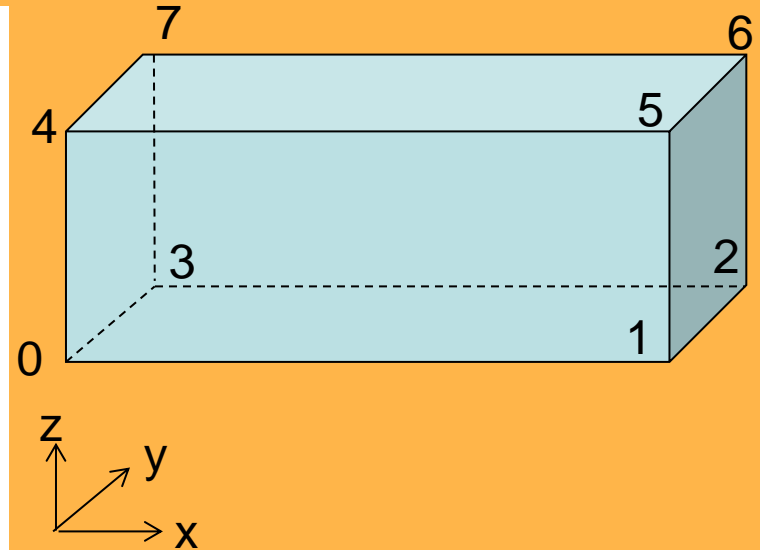
変更箇所は赤枠

```
10
17 convertToMeters 0.01; 単位変換 1m⇒0.01m
18
19 vertices 節点の座標 (x y z)
20 (
21   (-2 -1.5 0) //0
22   ( 4 -1.5 0) //1
23   ( 4  1.5 0) //2
24   (-2  1.5 0) //3
25   (-2 -1.5 0.1) //4
26   ( 4 -1.5 0.1) //5
27   ( 4  1.5 0.1) //6
28   (-2  1.5 0.1) //7
29 );
30
31 blocks 6面体指定
32 (
33   hex (0 1 2 3 4 5 6 7) (60 30 1) simpleGrading (1 1 1)
34 );
```

//の後はコメント

2次元問題とするためz方向
厚を1mmと小さくする

x,y,z方向の分割数

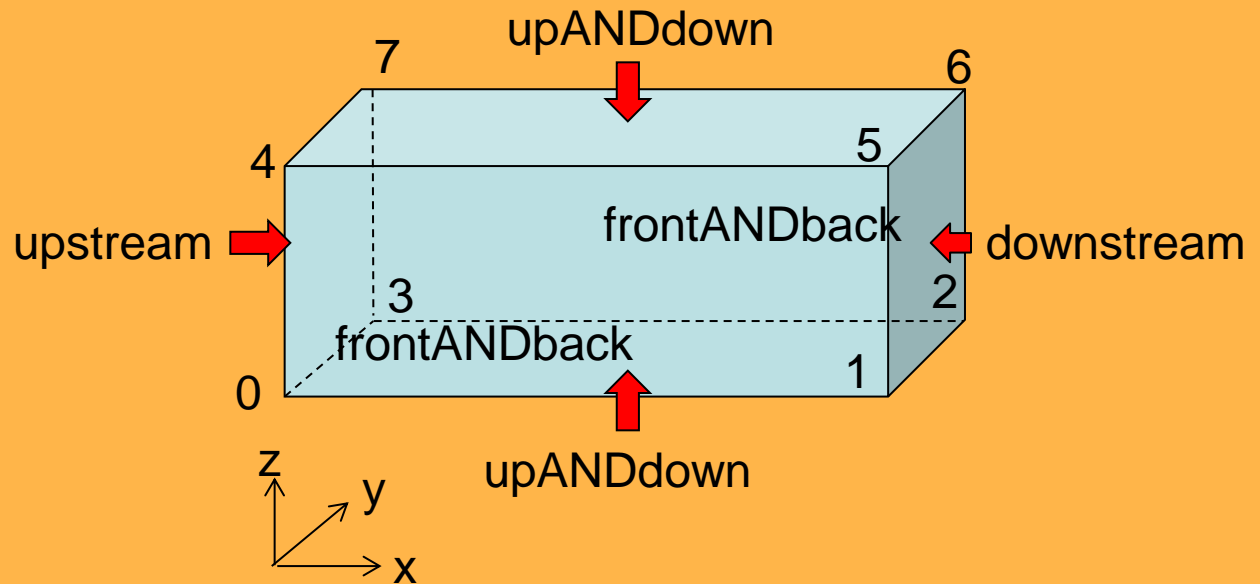


層流では $Re^{9/4}$ 個のセル数が目安
単位長さあたり $R^{3/4}$!
 $20^{3/4}=9.5$ なので、円柱直径10mm
を10分割する

1セル 1mm角
 $60 \times 30 \times 1 = 1800$ cells

基本境界 (Patch) 設定 (blockMeshDict)

```
blockMeshDict
40 boundary
41 (
42     upstream
43     {
44         type patch;
45         faces
46         (
47             (0 4 7 3)
48         );
49     }
50     downstream
51     {
52         type patch;
53         faces
54         (
55             (1 2 6 5)
56         );
57     }
58     upANDdown
59     {
60         type patch;
61         faces
62         (
63             (0 1 5 4)
64             (3 7 6 2)
65         );
66     }
67     frontANDback
68     {
69         type empty;
70         faces
71         (
72             (0 3 2 1)
73             (4 5 6 7)
74         );
75     }
76 );
77
```



2次元問題の時、計算しない面

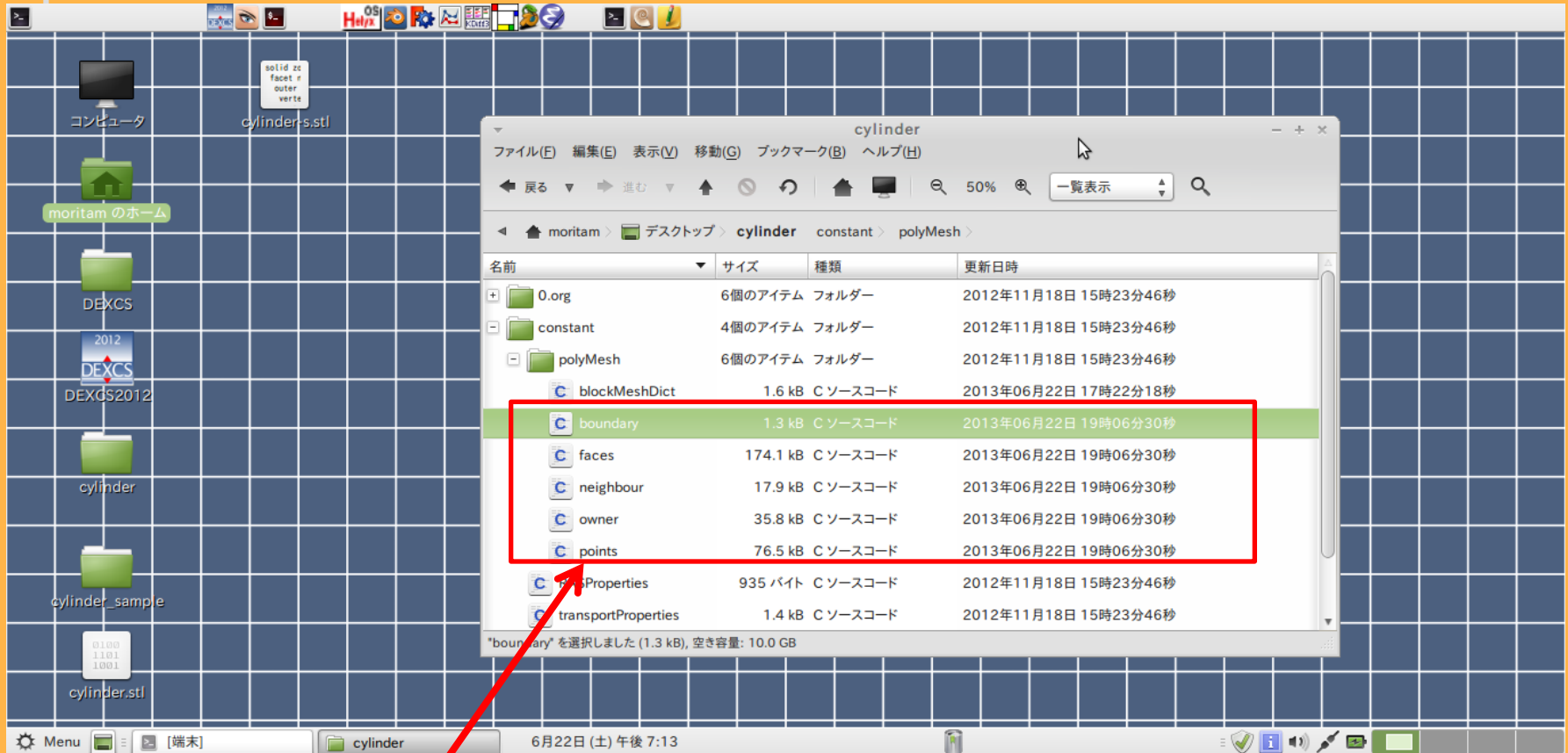
blockMesh作成

```
moritam@moritam-VirtualBox ~/Desktop $ cd cylinder
moritam@moritam-VirtualBox ~/Desktop/cylinder $ blockMesh

/*-----*¥
|=====|
|¥¥      | F ield      | OpenFOAM: The Open Source CFD Toolbox
|¥¥      | O peration  | Version: 2.1.x
|¥¥      | A nd        | Web: www.OpenFOAM.org
|¥¥/     | M anipulation|
|¥*-----*|
Build   : 2.1.x-fa2c23b05ebe
Exec    : blockMesh
Date    : Jun 22 2013
Time    : 19:06:30
Host    : "moritam-VirtualBox"
PID     : 2657
Case    : /home/moritam/Desktop/cylinder
nProcs  : 1
sigFpe  : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMast
allowSystemOperations : Disallowing user-supplied system call operations
```

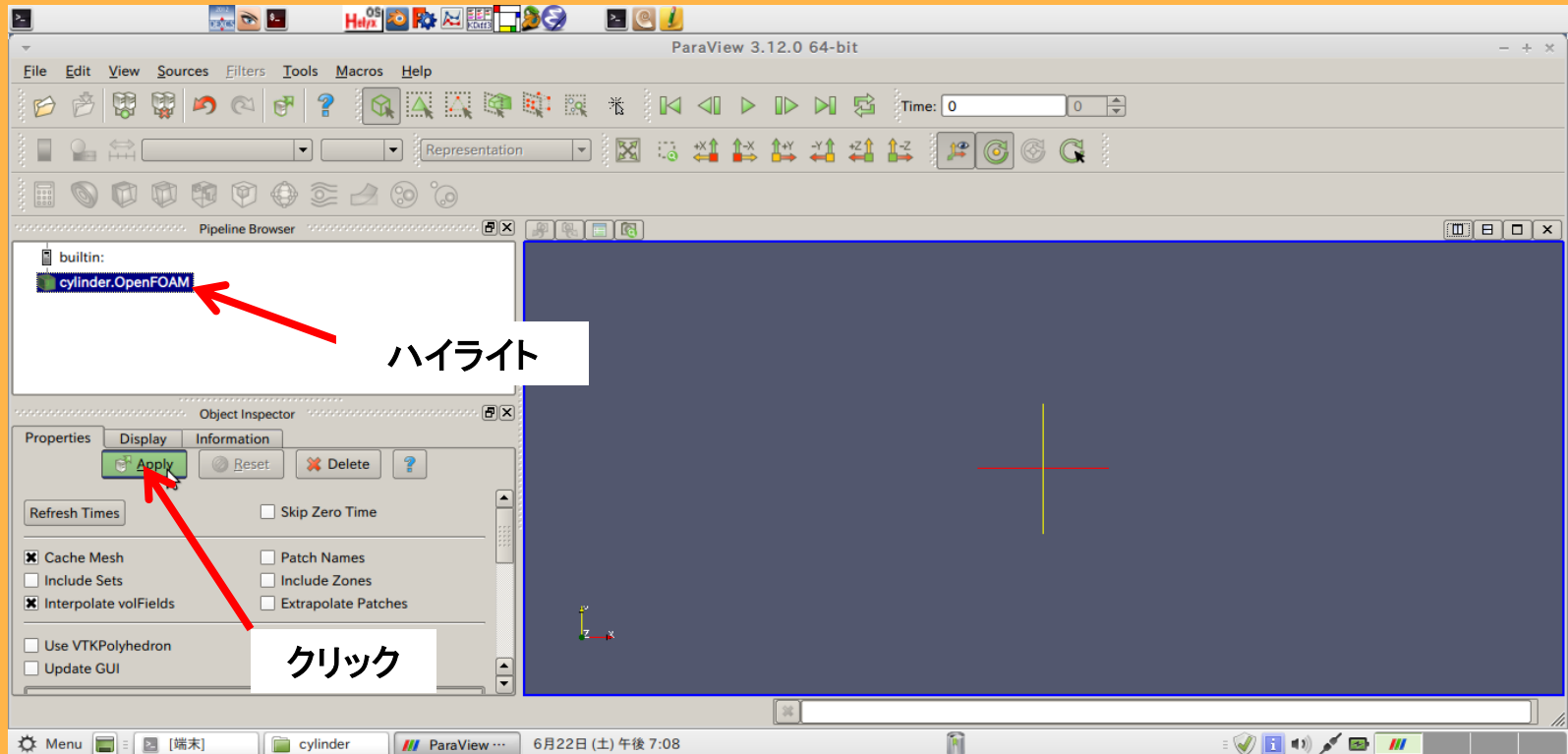
- ✓ コマンド実行フォルダへの移動
- ✓ `$cd cylinder`
- ✓ `blockMesh`コマンド実行
- ✓ `$blockMesh`

blockMesh実行後



変更や追加されたファイル

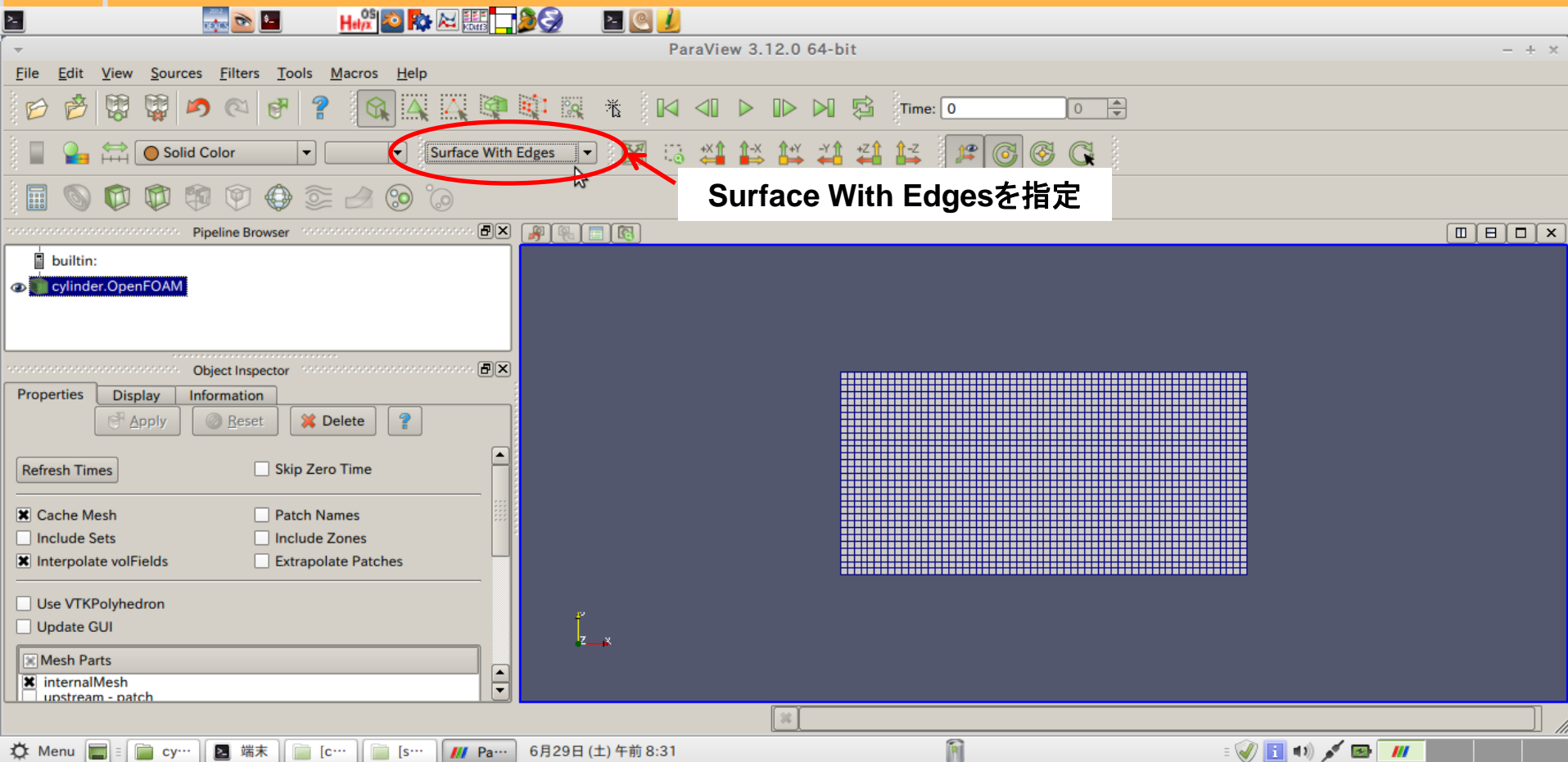
ParaFoamの起動(BlockMeshの確認)



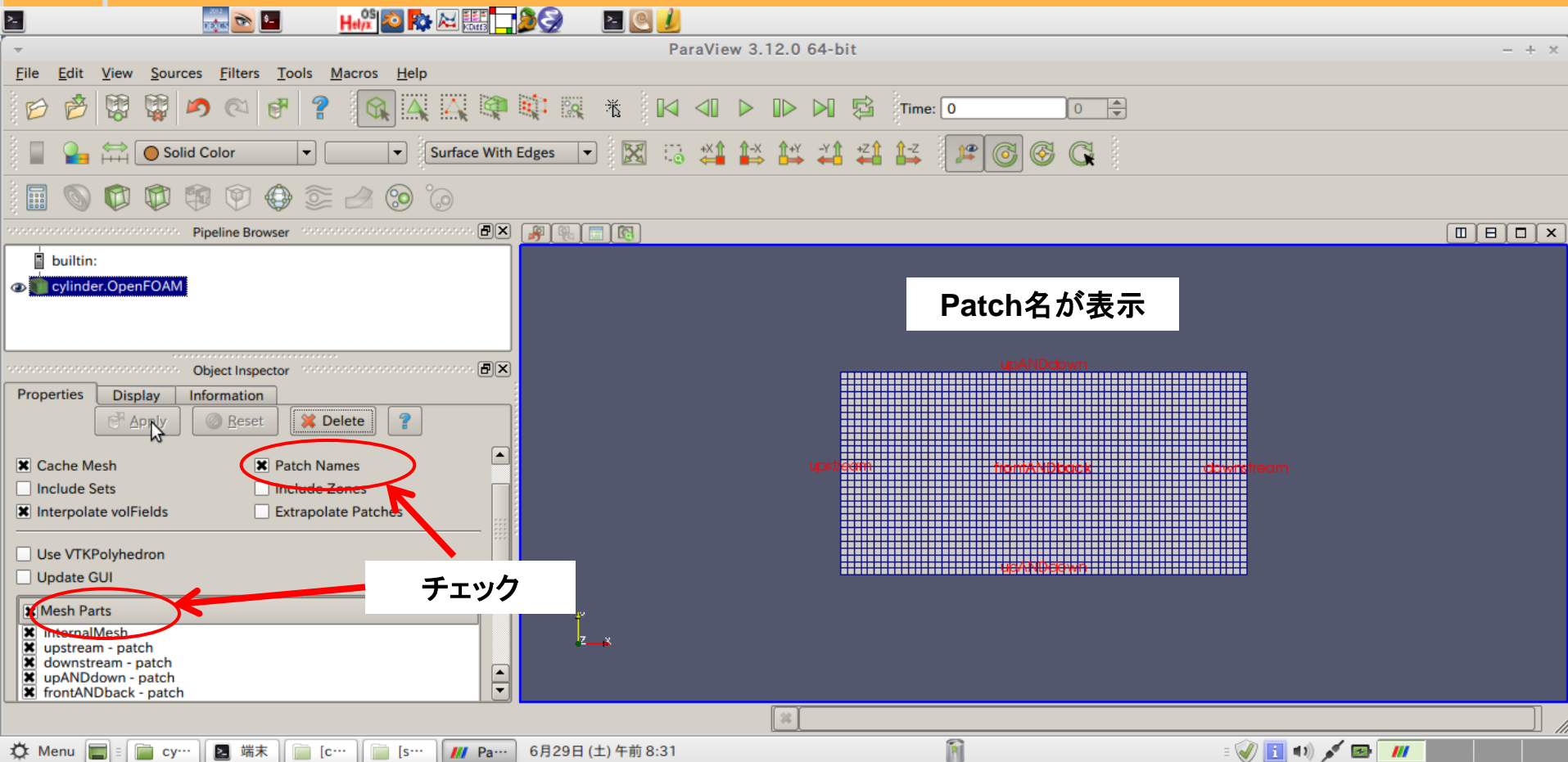
- ✓ ParaFoamの起動
- ✓ Applyをクリック

\$paraFoam

メッシュの確認



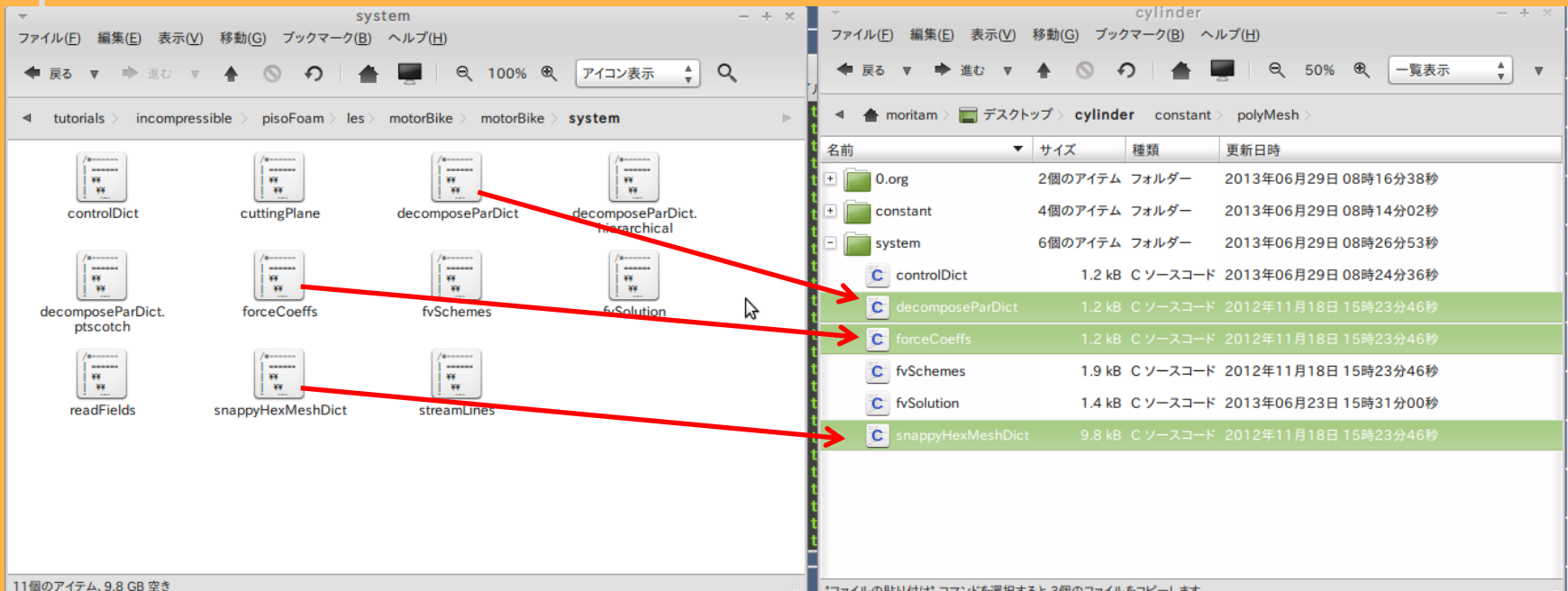
Patchの確認



snappyHexMesh作成

円柱周りのメッシュ作成

ベースファイルの取得



- ✓ `tutorials/incompressible/pisoFoam/les/motorBike/motorBike/system` から以下の3ファイルをコピー
 - ① `snappyHexMeshDict` (`snappyHexMesh`の制御ファイル)
 - ② `decomposeDict` (並列計算で使用。これがないと`snappy`でエラー)
 - ③ `forceCoeffs` (`Cd`値計算で使用。ついでにコピー)

snappyHexDictの変更

```
17 // Which of the steps to run
18 castellatedMesh true;
19 snap true;
20 addLayers false;
21
22
23 // Geometry. Definition of all surfaces. All surfaces are of class
24 // searchableSurface.
25 // Surfaces are used
26 // - to specify refinement for any mesh cell intersecting it
27 // - to specify refinement for any mesh cell inside/outside/near
28 // - to 'snap' the mesh boundary to the surface
29 geometry
30 {
31     cylinder-s.stl
32     {
33         type triSurfaceMesh;
34         name cylinder;
35     }
```

1段階(O):表面に適合したメッシュ除去

2段階(O):表面メッシュのスナップ(平滑化)

3段階(x):表面へのレイヤー挿入

円柱ファイルの名称

ファイルタイプ:stl

表面の名前指定:cylinder

snappyHexMeshDictの変更

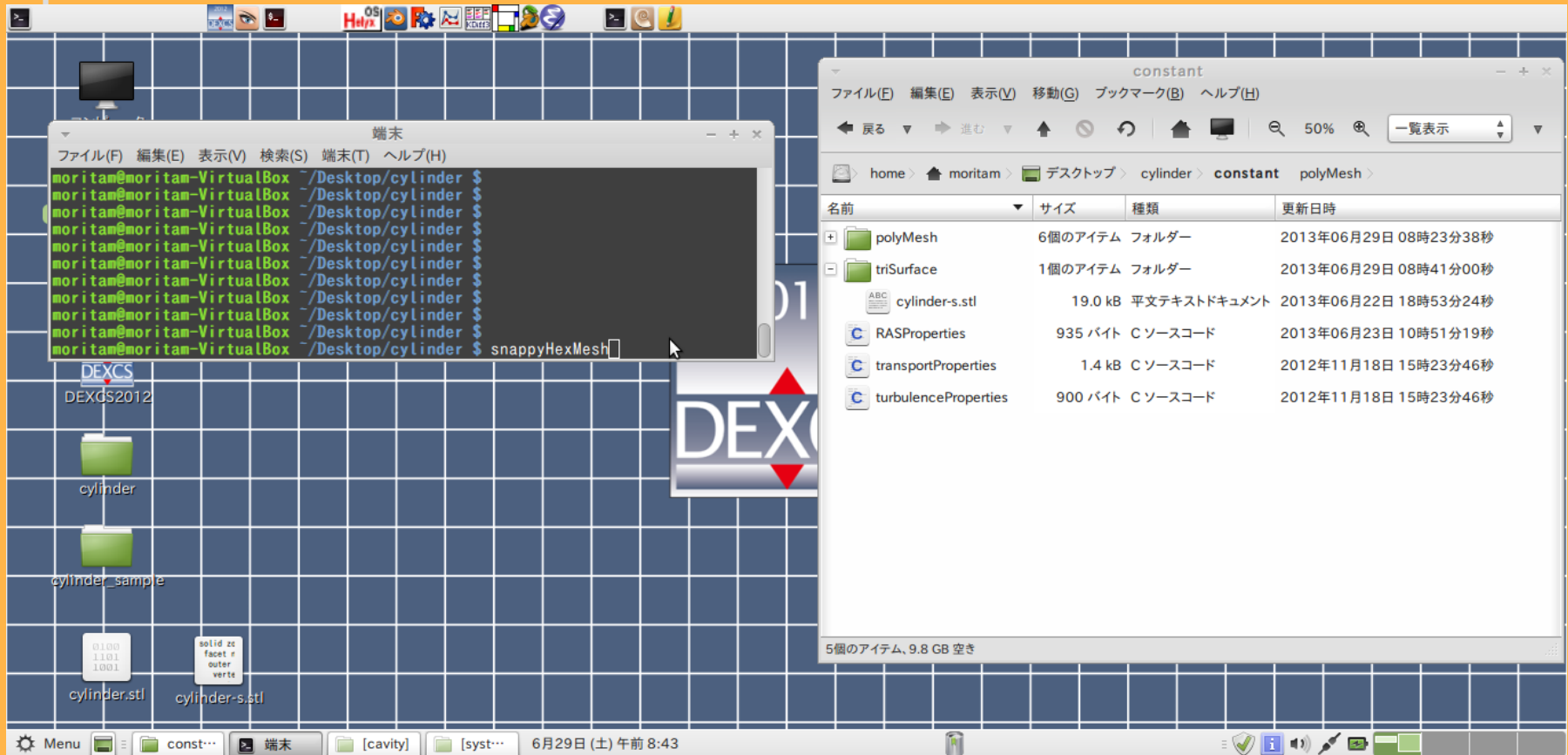
```
110
111 refinementSurfaces
112 {
113     cylinder 細分化したい表面の名称
114     {
115         // Surface-wise min and max refinement level
116         level (0 0); 細分化の程度（最小、最大）
117     } 数が0⇒1⇒2⇒...大きくなるほど細分化
118 } 最小レベルは全ての表面に適用
119 // Resolve sharp angles 最大レベルはresolveFeatureAngle 以上の角度の表面に適用
120 resolveFeatureAngle 60;
121
122
```

```
137 refinementRegions
138 {
139     // refinementBox
140     {
141         mode inside;
142         levels ((1E15 1));
143     }
144 }
145
146 // Mesh selection
147 // -----
148
149 // After refinement patches get added for all refinementSurfaces and
150 // all cells intersecting the surfaces get put into these patches. The
151 // section reachable from the locationInMesh is kept.
152 // NOTE: This point should never be on a face, always inside a cell, even
153 // after refinement.
154 locationInMesh (-0.01 0 0.0005);
155
156
```

細分化したいBox領域の細分化
今回使わないのでコメントアウト

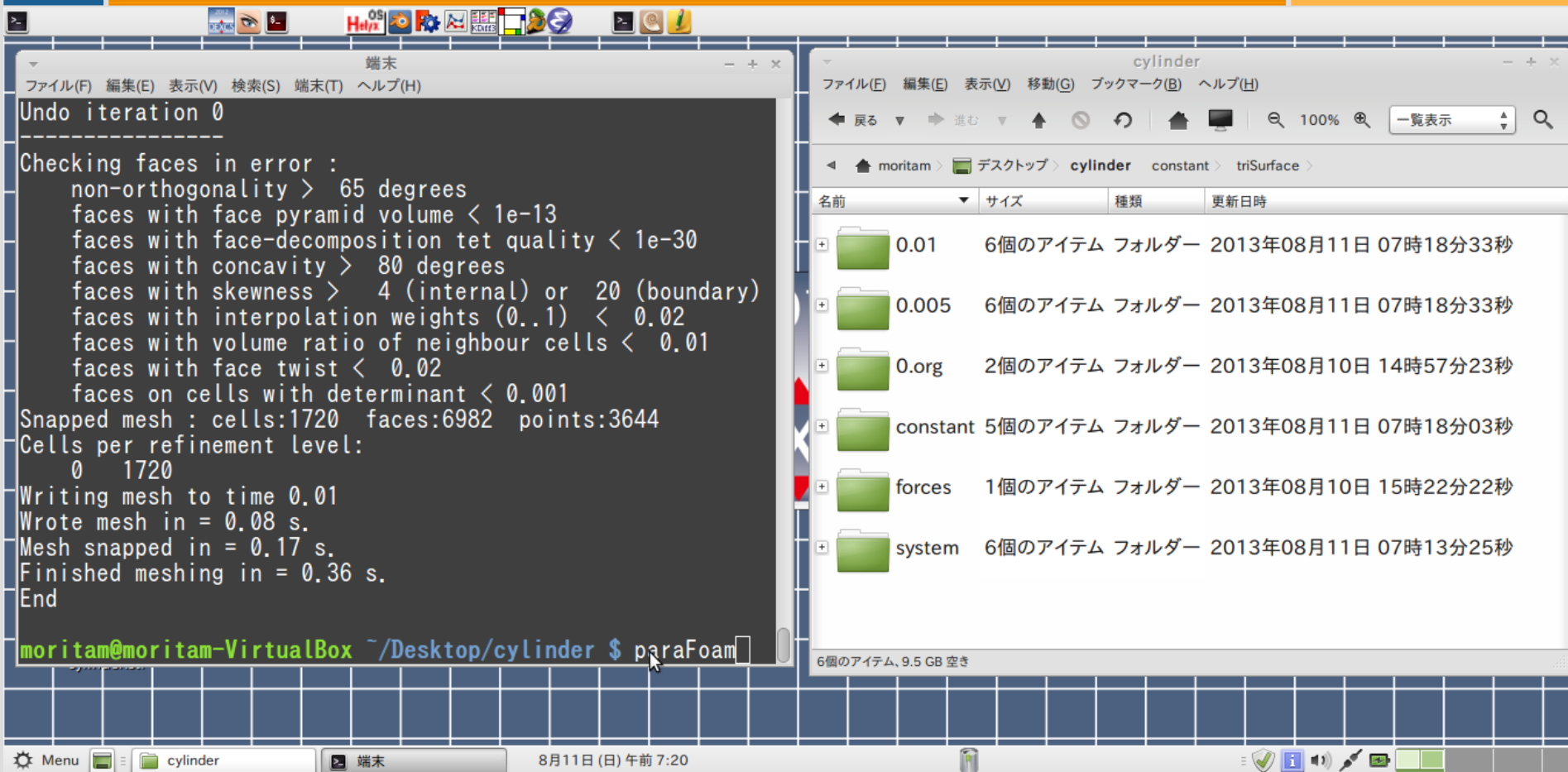
Snappyを適用する領域を指定。今回は円柱の外部の座標を選ぶ。セルの表面の座標は好ましくない。但し、今回の点はセル表面であるが、認識できた。

snappyHexMesh実行



✓ **snappyHexMesh実行** \$snappyHexMesh

snappyHexMesh実行



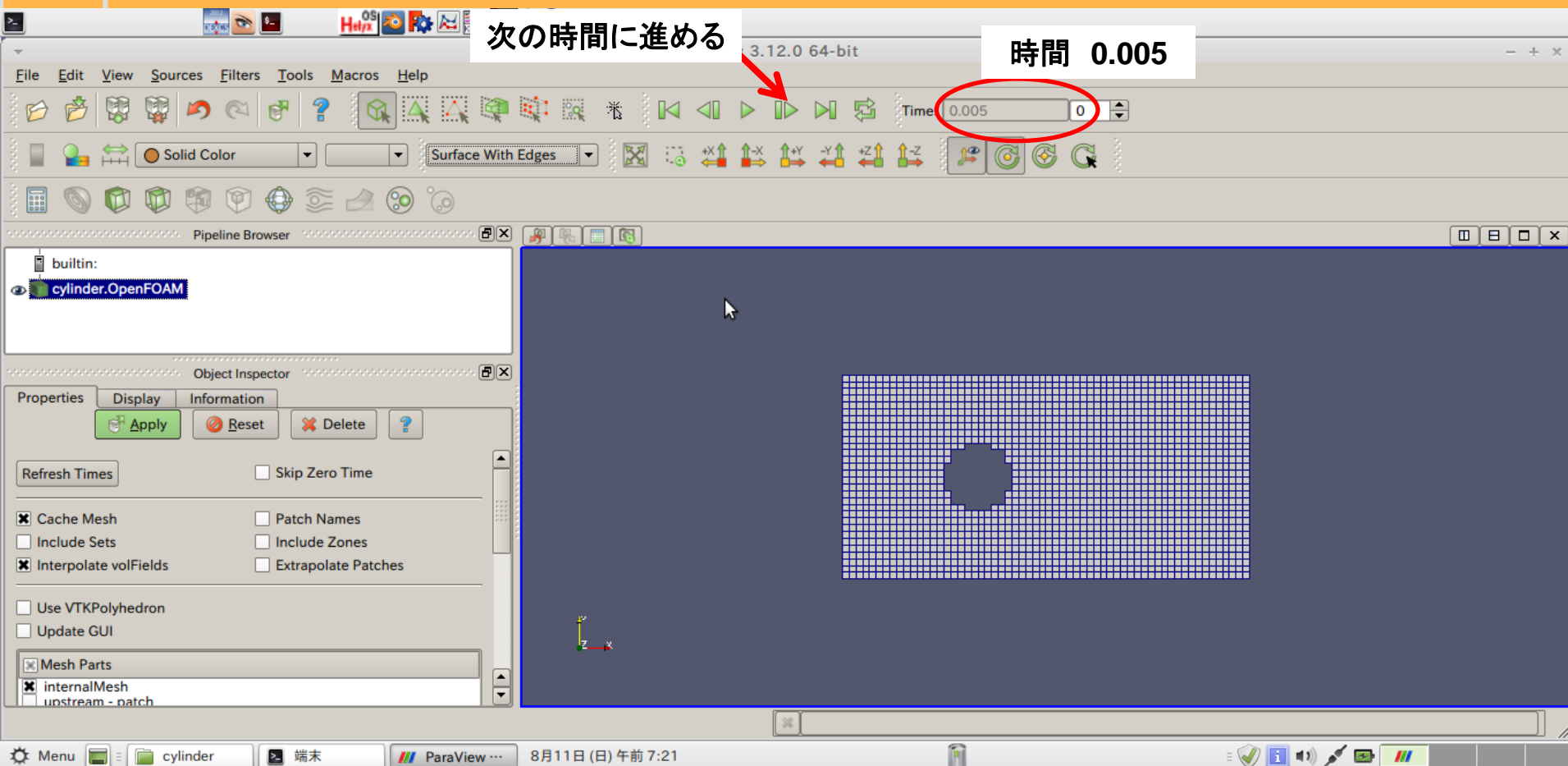
✓ snappy成功 cell数は1720

✓ 0.005、0.01ディレクトリが生成

(後に説明するcotrolDictの時間ディレクトリ作成間隔が0.005secの場合)

2013/8/10

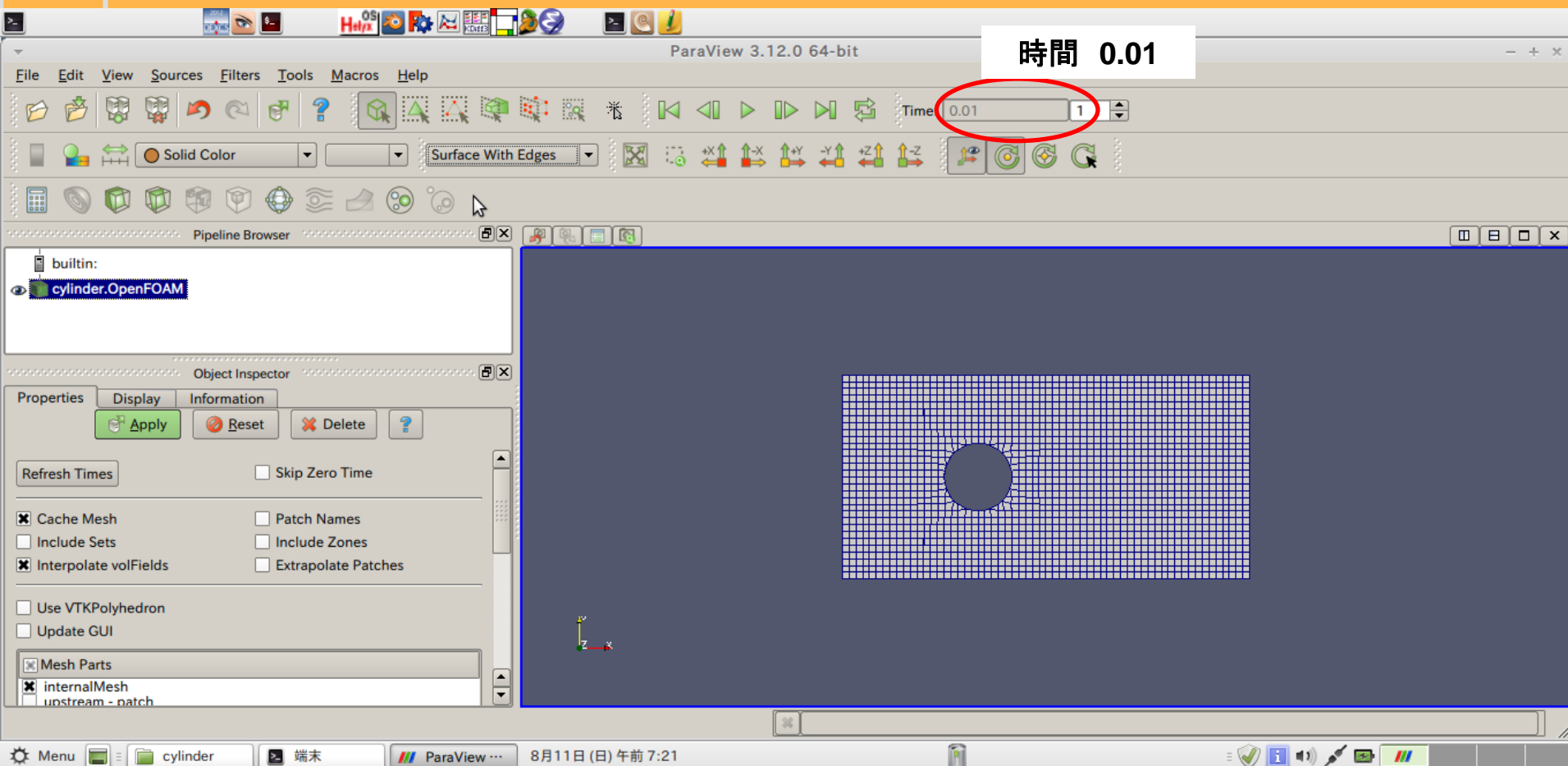
メッシュの確認(第1段階)



- ✓ \$paraFoamを起動
- ✓ 円柱部分のセルが除去されている
- ✓ 表面は階段状

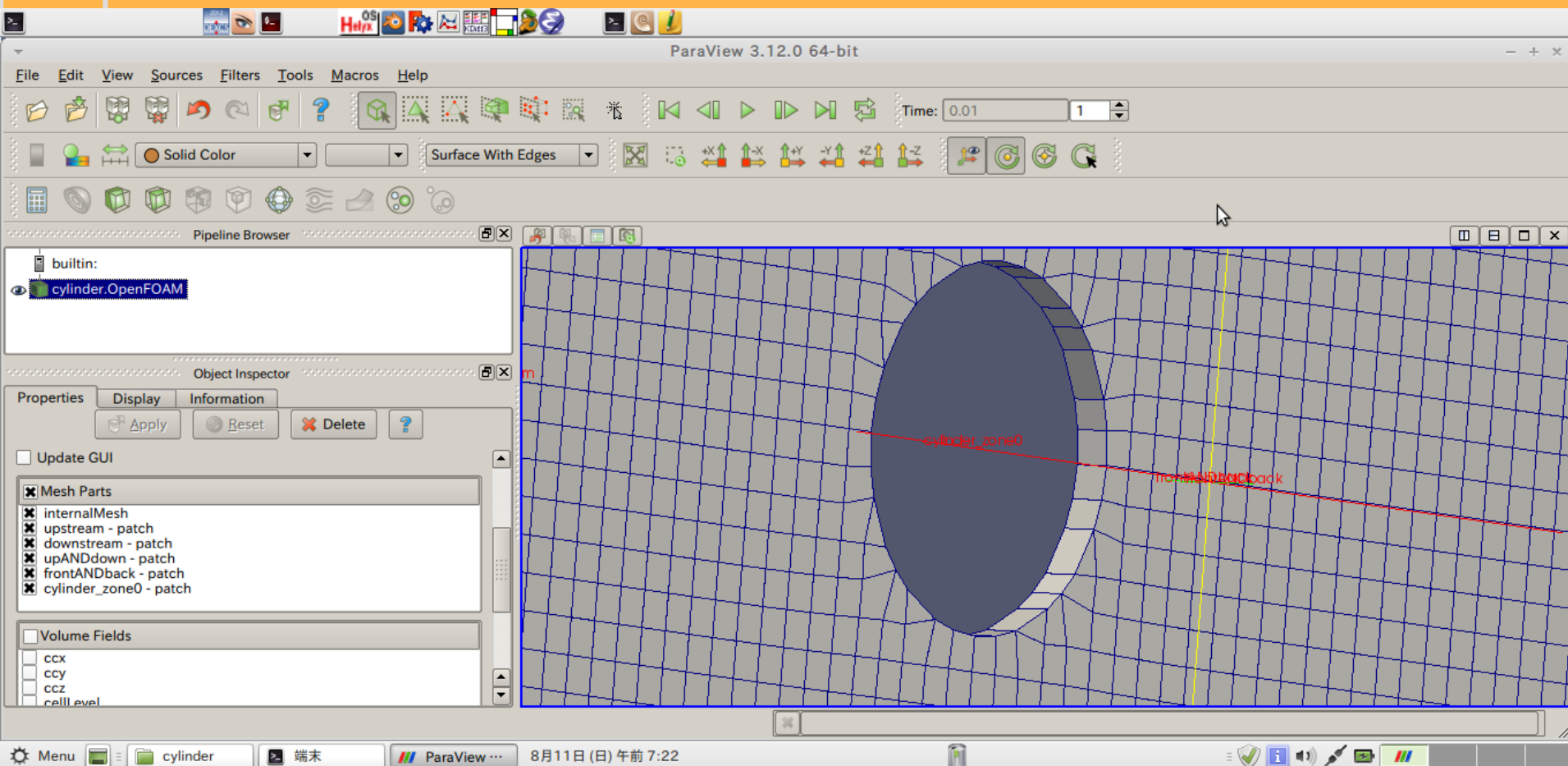
2013/8/10

メッシュの確認（第2段階）



✓ 円柱表面の平滑化

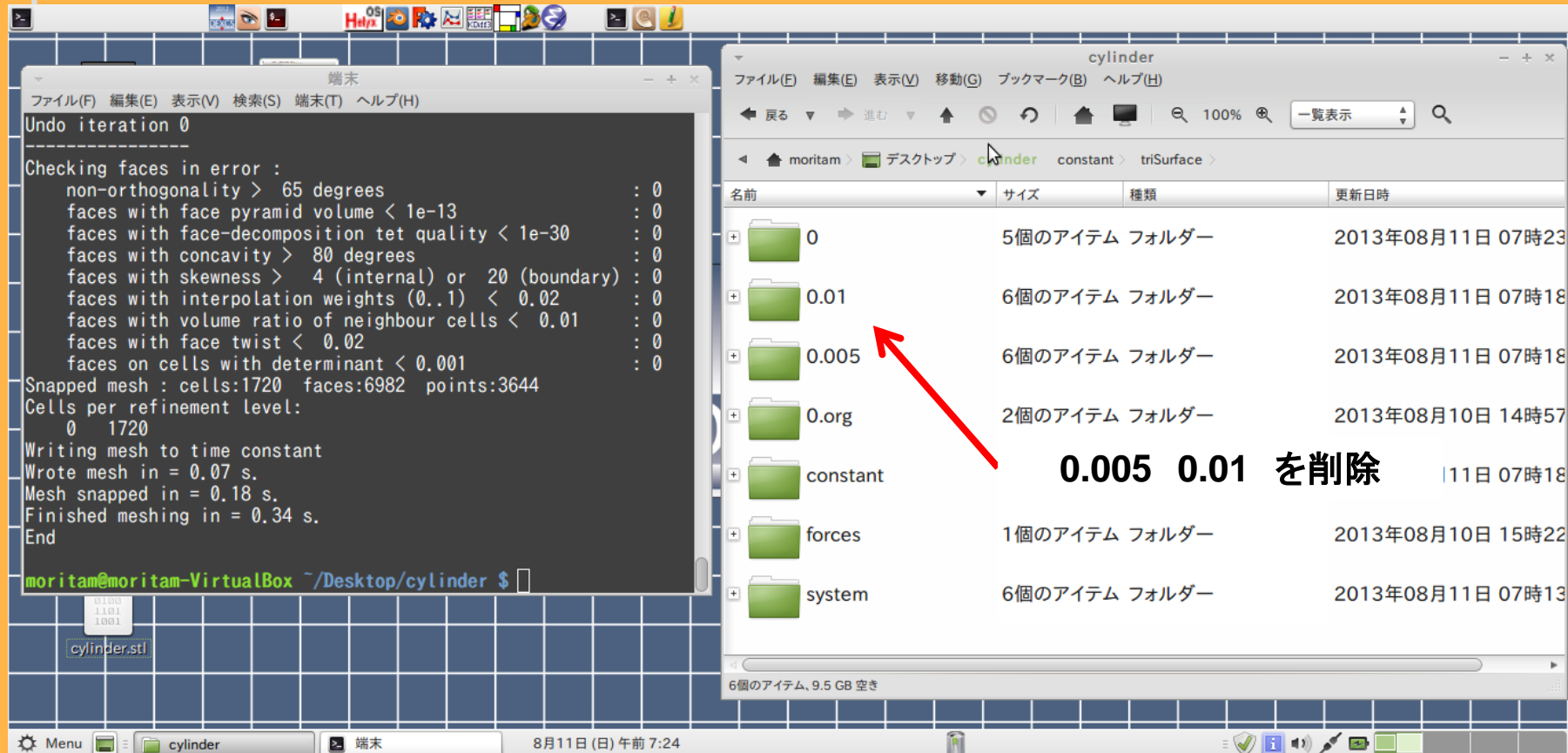
メッシュ確認



- ✓ 円柱側面は1層だけ切れている。(2次元問題OK)
- ✓ 円柱側面のPatchはcylinder_zone0の名前が付けられている。

2013/8/10

snappyHexMesh(再)



✓ 生成した0.01ディレクトリのメッシュを使って0ディレクトリを作っても良いが別の方法をトライ。一回で0ディレクトリが生成

✓ `$snappyHexMesh -overwrite`

2013/8/10

✓ `$rm -r 0.005 0.01` (0.005 0.01を削除)

0ディクショナリ作成

初期値・境界値の設定

U

constant/polyMesh/boundary (Patch名とtype等を記述)

0.org/U

boundary *	17 dimensions	[0 1 -1 0 0 0 0]; 次元(kg m s K mol A cd)=(m/s)
19 {	18 internalField	uniform (2.0E-2 0 0); 領域初期値 全域2cm/s
20 upstream	19 boundaryField	
21 {	20 {	
22 type patch;	21 upstream	
23 nFaces 30;	22 {	
24 startFace 3330;	23 type	fixedValue;
25 }	24 value	uniform (2.0E-2 0 0); 固定値 上流2cm/s
26 downstream	25 }	
27 {	26 downstream	チュートリアルでよく使っているで使用
28 type patch;	27 {	
29 nFaces 30;	28 type	inletOutlet; 流出条件(流入あり)
30 startFace 3360;	29 inletValue	uniform (2.0E-2 0 0) 流入時 速度2cm/s
31 }	30 value	\$internalField; 流出時 内部メッシュ値
32 upANDdown	31 }	
33 {	32 upANDdown	
34 type patch;	33 {	
35 nFaces 120;	34 type	inletOutlet; 流出条件(流入あり)
36 startFace 3390;	35 inletValue	uniform (2.0E-2 0 0) 流入時 速度2cm/s
37 }	36 value	\$internalField; 流出時 内部メッシュ値
38 frontANDback	37 }	
39 {	38 frontANDback	
40 type empty;	39 {	
41 nFaces 3440;	40 type	empty; 2次元問題条件
42 startFace 3510;	41 }	
43 }	42 cylinder_zone0	
44 cylinder_zone0	43 {	
45 {	44 type	fixedValue;
46 type wall;	45 value	uniform (0 0 0); 固定値 粘着条件 43
47 nFaces 32;	46 }	
48 startFace 6950;	47 }	
49 }		
50 }		

p=(p/p)

0.org/p

```

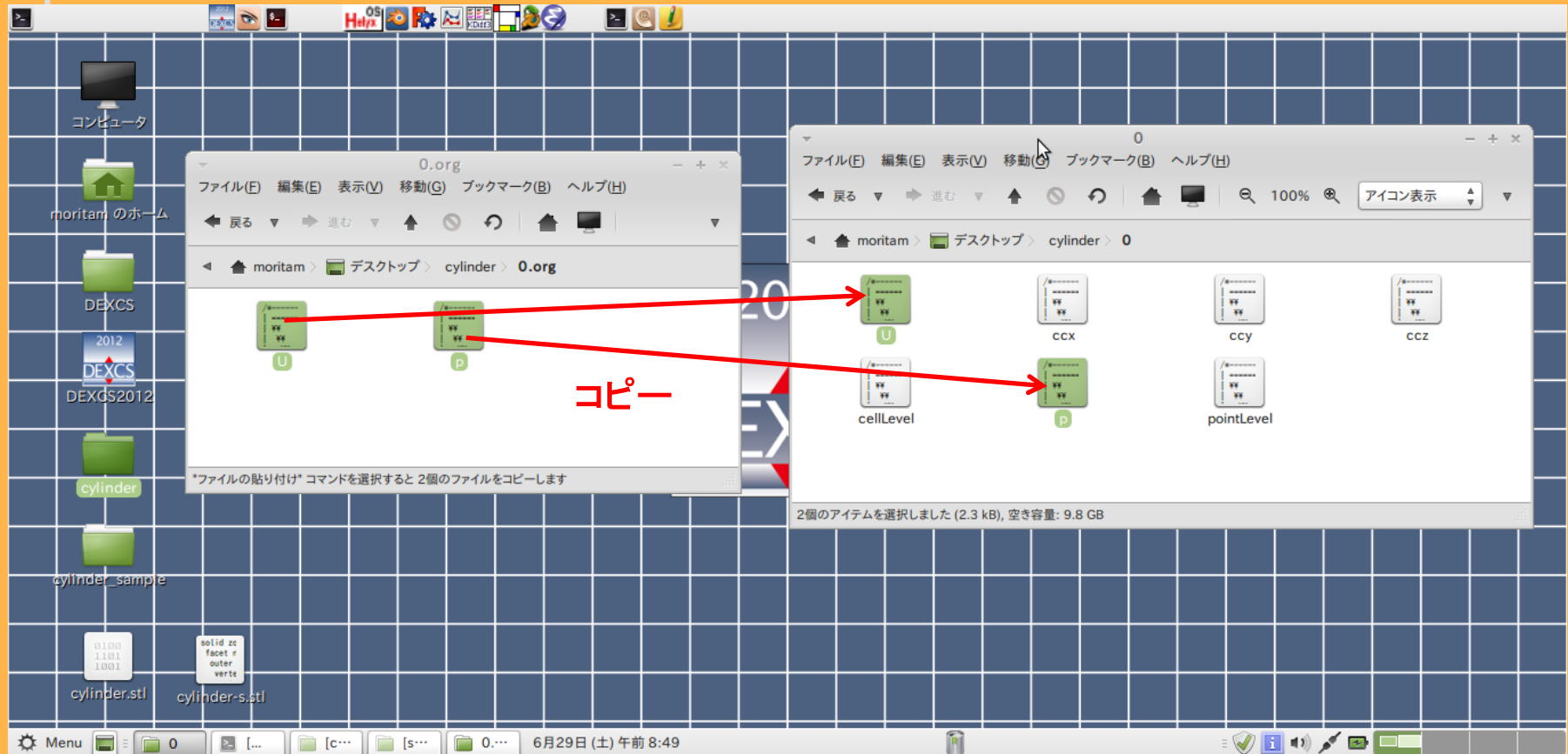
17 dimensions      [0 2 -2 0 0 0 0]; 次元=(m²/s²)
18 internalField    uniform 0;
19 boundaryField
20 {
21     upstream
22     {
23         type      fixedValue;
24         value      uniform 0;    p=0
25     }
26     downstream
27     {
28         type      fixedValue;
29         value      uniform 0;    p=0
30     }
31     upANDdown
32     {
33         type      fixedValue;
34         value      uniform 0;    p=0
35     }
36     frontANDback
37     {
38         type      empty;        2次元問題条件
39     }
40     cylinder_zone0
41     {
42         type      zeroGradient;  $\frac{\partial p}{\partial n} = 0$ 
43     }
44 }

```

2013/8/10

あまり適切ではないが、選択できる条件では、これが最もよい(付録参照)

0ディクショナリ作成



✓ 0ディクショナリにはU、pファイルがないので0.orgからコピー

Constantディクショナリ のその他ファイルの設定

物性値と流体モデル選定

物性値と解析モデル

turbulencePropertiesファイル(変更なし)

```
16 // *****
17
18 simulationType  RASModel;
19
```

乱流モデルの選択

RASmodel (Reynolds平均モデル)

LESmodel (Large Eddy Simulation モデル)

transportPropertiesファイル(変更なし)

```
17
18 transportModel  Newtonian;
19
20 nu 動粘性係数      nu [ 0 2 -1 0 0 0 0 ] 1e-05;
21                                     [m²/s]
22 CrossPowerLawCoeffs
```

輸送係数モデル選択

Newtonian (ニュートン流体)

CrossPowerLawCoefss (非ニュートン流体)

など

2013/8/10

RASPropertiesファイル

```
C RASProperties ✖
1  /*-----
2  |=====
3  |  ¥¥          F i e l d
4  |  ¥¥          O p e r a t i o n
5  |  ¥¥          A n d
6  |  ¥¥/         M a n i p u l a t i o n
7  /*-----
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     location      "constant";
14     object        RASProperties;
15 }
16 // *****
17
18 RASModel          laminar; 層流
19
20 turbulence        off;
21
22 printCoeffs       on;
23
24
25 // *****
```

systemディクショナリ のファイル設定

計算スキームと計算制御の設定

controlDict設定

```
18 application      pisoFoam;  
19 startFrom        startTime;  
20 startTime        0;  
21 stopAt           endTime;  
22 endTime          15;  
23 deltaT           0.001;  
24 writeControl     timeStep;  
25 writeInterval    1000;  
26 purgeWrite       0;  
27 writeFormat      ascii;  
28 writePrecision   6;  
29 writeCompression off;  
30 timeFormat       general;  
31 timePrecision    6;  
32 runTimeModifiable true;  
33 functions  
34 {  
35     #include "forceCoeffs"  
36 }
```

⇒計算ソルバー

⇒startTimeの設定時間から計算開始

⇒計算開始時間

⇒endTimeの設定時間に計算終了

⇒計算終了時間(試行錯誤で決定)

⇒計算時間ステップ(クーラン数で決定)

⇒writeIntervalごとに時刻ディクショナリ生成

⇒1000ステップ、1秒ごと

⇒時刻ディレクトリ最大数:制限なし

⇒データファイルフォーマット選択

⇒writeFormatでの有効桁数

⇒データ圧縮指定

⇒timeFormatのフォーマット選択

⇒timeFormat の設定で用いる指数

⇒各ディクショナリのyes/noスイッチ

⇒物体に係る力の係数計算用の関数

forceCoeffs

```
9 forces
10 {
11     type            forceCoeffs;
12     functionObjectLibs ( "libforces.so" );
13     outputControl    timeStep;
14     outputInterval    100;
15
16     patches          ( "cylinder zone0" ); 円柱Patch
17     pName             p;
18     UName             U;
19     rhoName           rhoInf; // Indicates incompressible
20     log               true;
21     密度 rhoInf       1000; [kg/m³] // Redundant for incompressible
22     liftDir           (0 1 0); 揚力方向 y
23     dragDir           (1 0 0); 抗力方向 x
24     CofR              (0 0 0); // Axle midpoint on ground 物体中心
25     pitchAxis         (0 0 1); ピッチモーメントの回転軸 z
26     magUInf           0.02; 上流での流速 [m/s]
27     lRef              0.01; // Wheelbase length 代表寸法 [m]
28     Aref              1.0E-5; // Estimated 射影面積 [m²]
29 }
```

fvscheme (変更なし)

```
18 ddtSchemes
19 {
20     default          Euler;
21 }
22 gradSchemes
23 {
24     default          Gauss linear;
25     grad(p)           Gauss linear;
26     grad(U)           Gauss linear;
27 }
28 divSchemes
29 {
30     default          none;
31     div(phi,U)        Gauss limitedLinearV 1;
32     div((nuEff*dev(T(grad(U)))) Gauss linear;
33 }
34 laplacianSchemes
35 {
36     default          none;
37     laplacian(nuEff,U) Gauss linear corrected;
38     laplacian((1|A(U)),p) Gauss linear corrected;
39 }
```

fvscheme (変更なし)

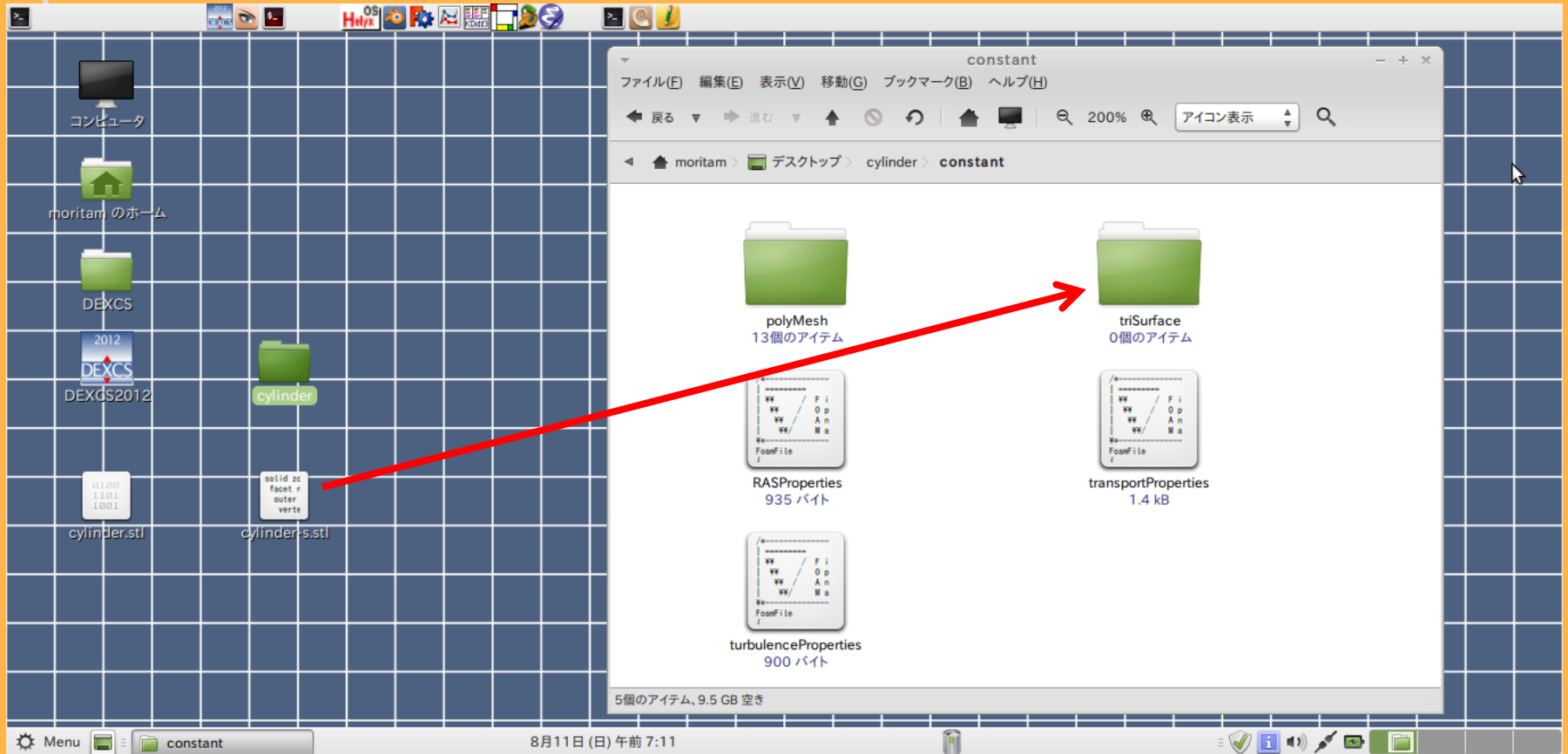
```
40 interpolationSchemes
41 {
42     default          linear;
43     interpolate(U)    linear;
44 }
45 snGradSchemes
46 {
47     default          corrected;
48 }
49 fluxRequired
50 {
51     default          no;
52     p                 ;
53 }
```


Fvsolution(変更なし)

```
18 solvers
19 {
20     p
21     {
22         solver          PCG;
23         preconditioner  DIC;
24         tolerance       1e-06;
25         relTol          0.1;
26     }
27     pFinal
28     {
29         solver          PCG;
30         preconditioner  DIC;
31         tolerance       1e-06;
32         relTol          0;
33     }
34     U
35     {
36         solver          PBiCG;
37         preconditioner  DILU;
38         tolerance       1e-05;
39         relTol          0;
40     }
41 }
42
43 PISO
44 {
45     nCorrectors          2;
46     nNonOrthogonalCorrectors 0;
47     pRefCell              0;
48     pRefValue             0;
49 }
```

STLファイルの格納

triSurfaceへ格納

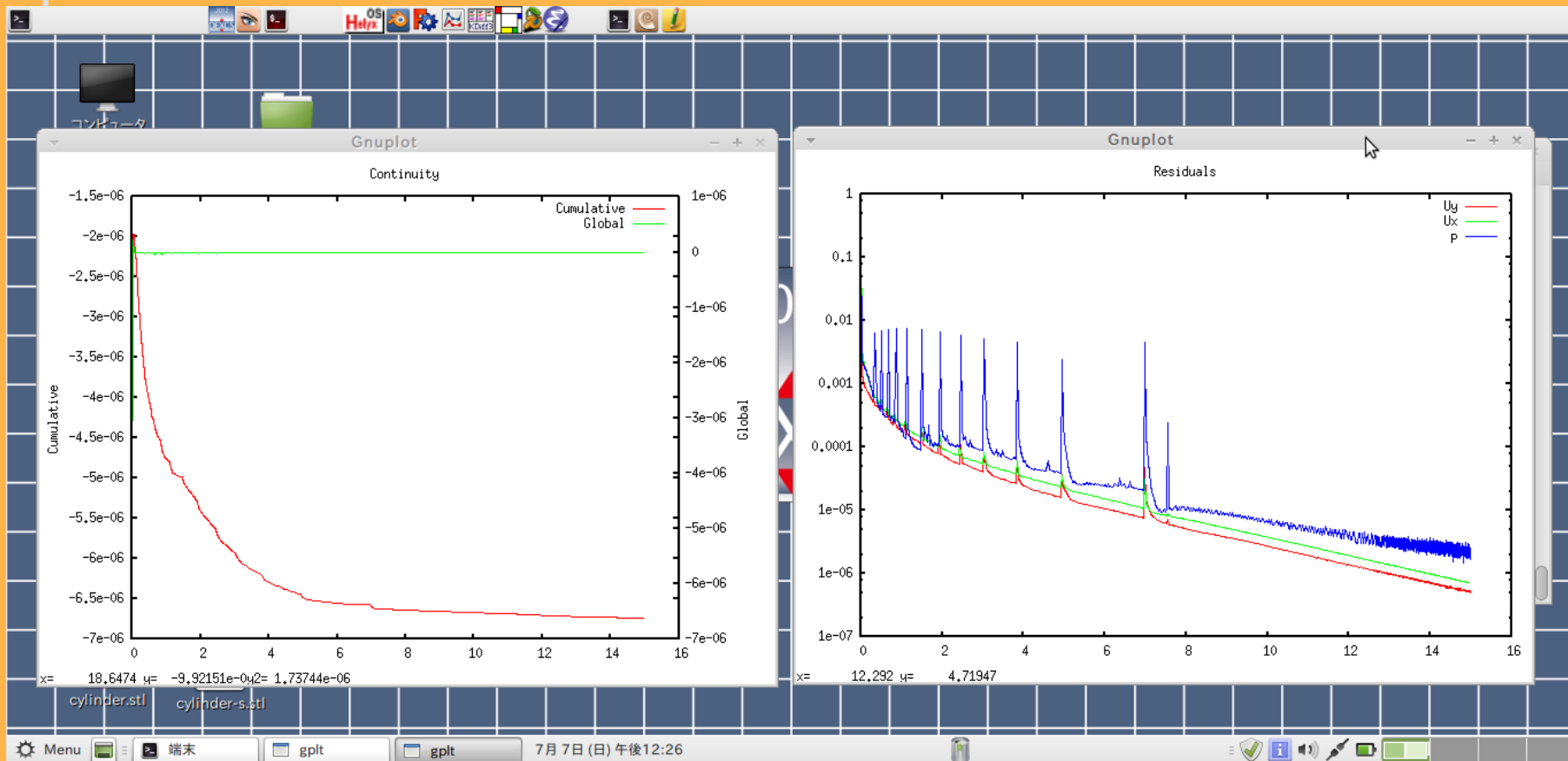


- ✓ snappyHexMeshで指定したcylinder-s.stlについて
constant/triSurfaceディクショナリを見に行く
- ✓ constantにtriSurfaceを作成
- ✓ Desktopのcylinder-s.stlをtriSurfaceに移動

2013/8/10

計算実行

pisFoamの実行



- ✓ `$pyFoamPlotRunner.py pisoFoam`
- ✓ `pyFoamPlotRunner.py`で残差等がモニターできる。

計算終了時の画面

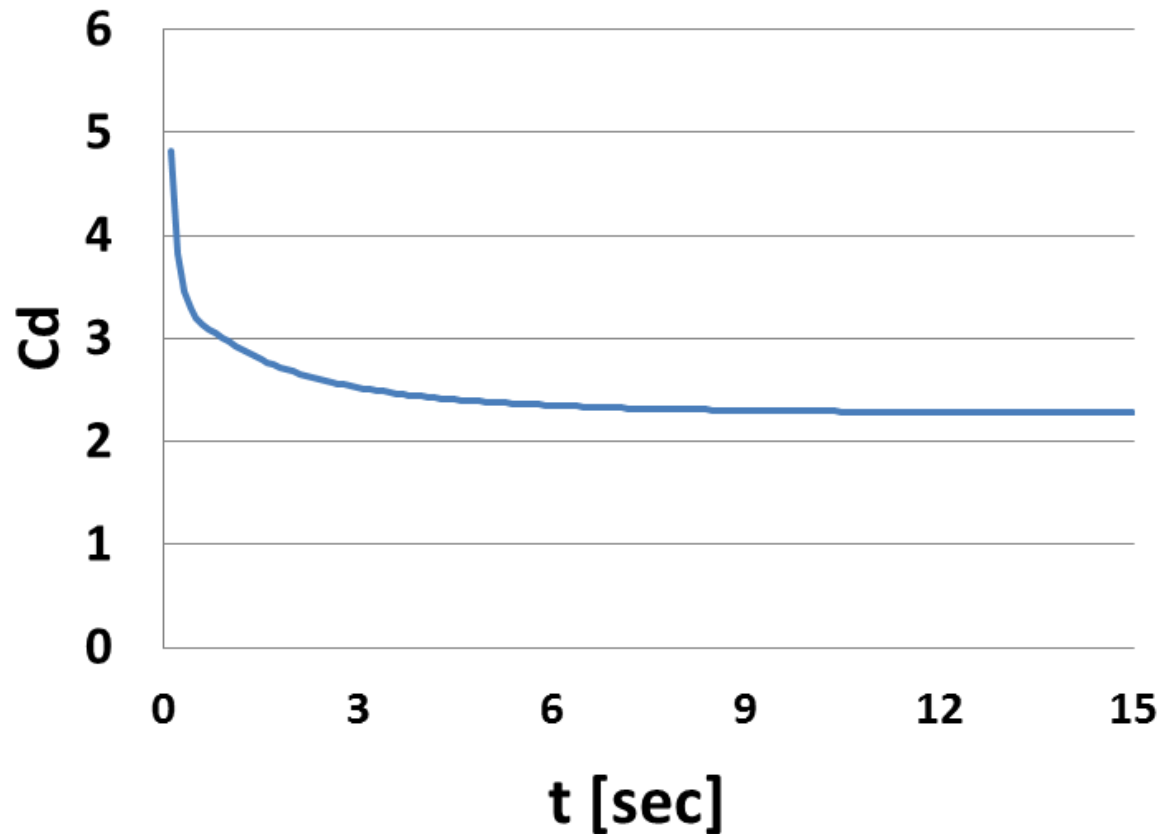
```
端末
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
DILUPBiCG: Solving for Ux, Initial residual = 7.14331e-07, Final residual = 7.14331e-07, No Iterations 0
DILUPBiCG: Solving for Uy, Initial residual = 5.02723e-07, Final residual = 5.02723e-07, No Iterations 0
DICPCG: Solving for p, Initial residual = 1.61101e-06, Final residual = 5.28806e-07, No Iterations 1
time step continuity errors : sum local = 2.33894e-11, global = -1.1056e-11, cumulative = -6.73347e-06
DICPCG: Solving for p, Initial residual = 6.26445e-07, Final residual = 6.26445e-07, No Iterations 0
time step continuity errors : sum local = 2.7708e-11, global = -1.30594e-11, cumulative = -6.73348e-06
ExecutionTime = 72.8 s  ClockTime = 137 s  計算時間

forceCoeffs output:
Cd = 2.2784      抗力係数   (実験値は2.0程度)
Cl = 2.58734e-07 揚力係数
Cm = -1.32201e-09 ピッチモーメント係数
Cl(f) = 1.30689e-07
Cl(r) = 1.28045e-07

End

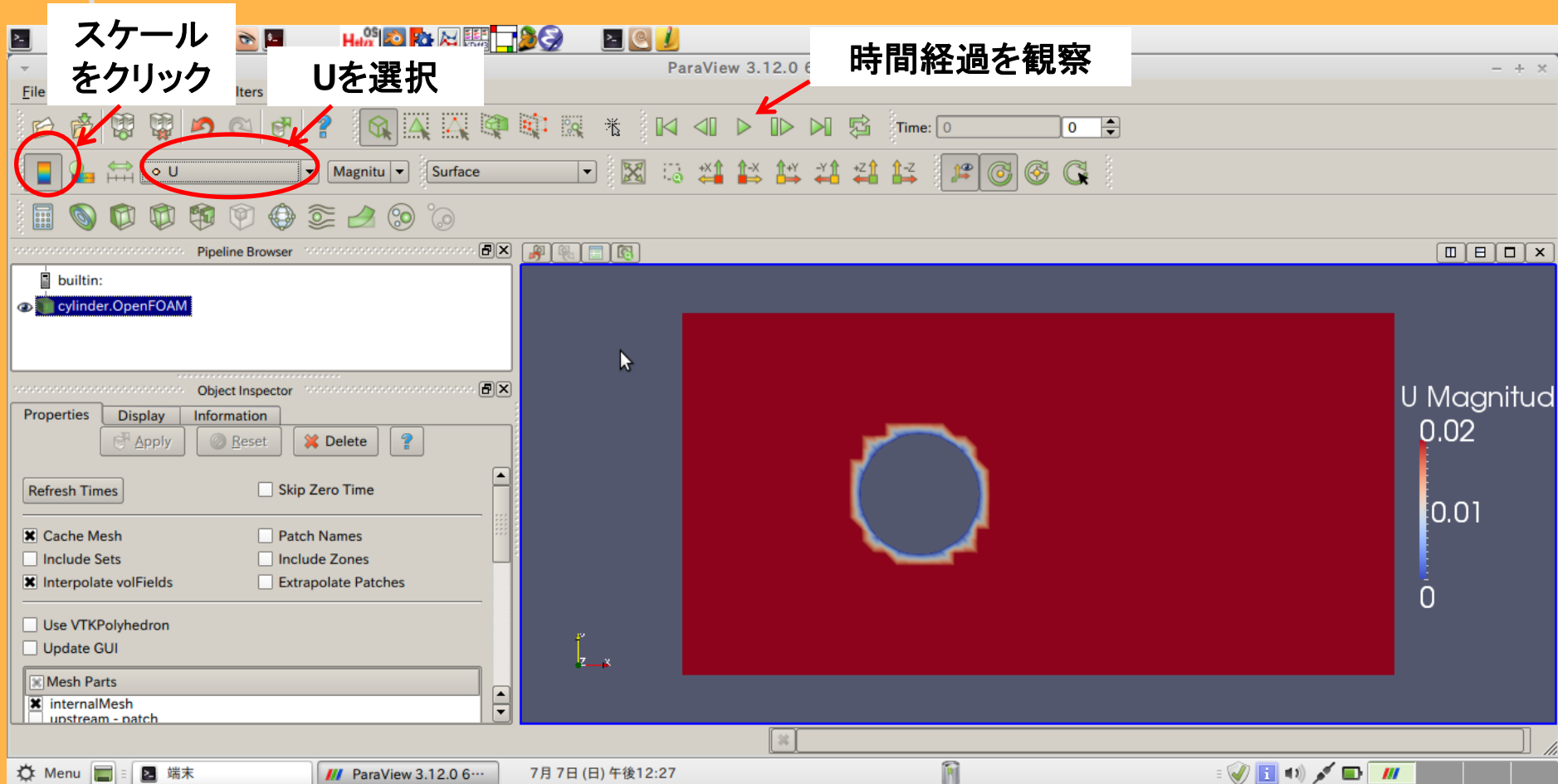
moritan@moritan-VirtualBox ~/Desktop/cylinder $
```

Cdの履歴



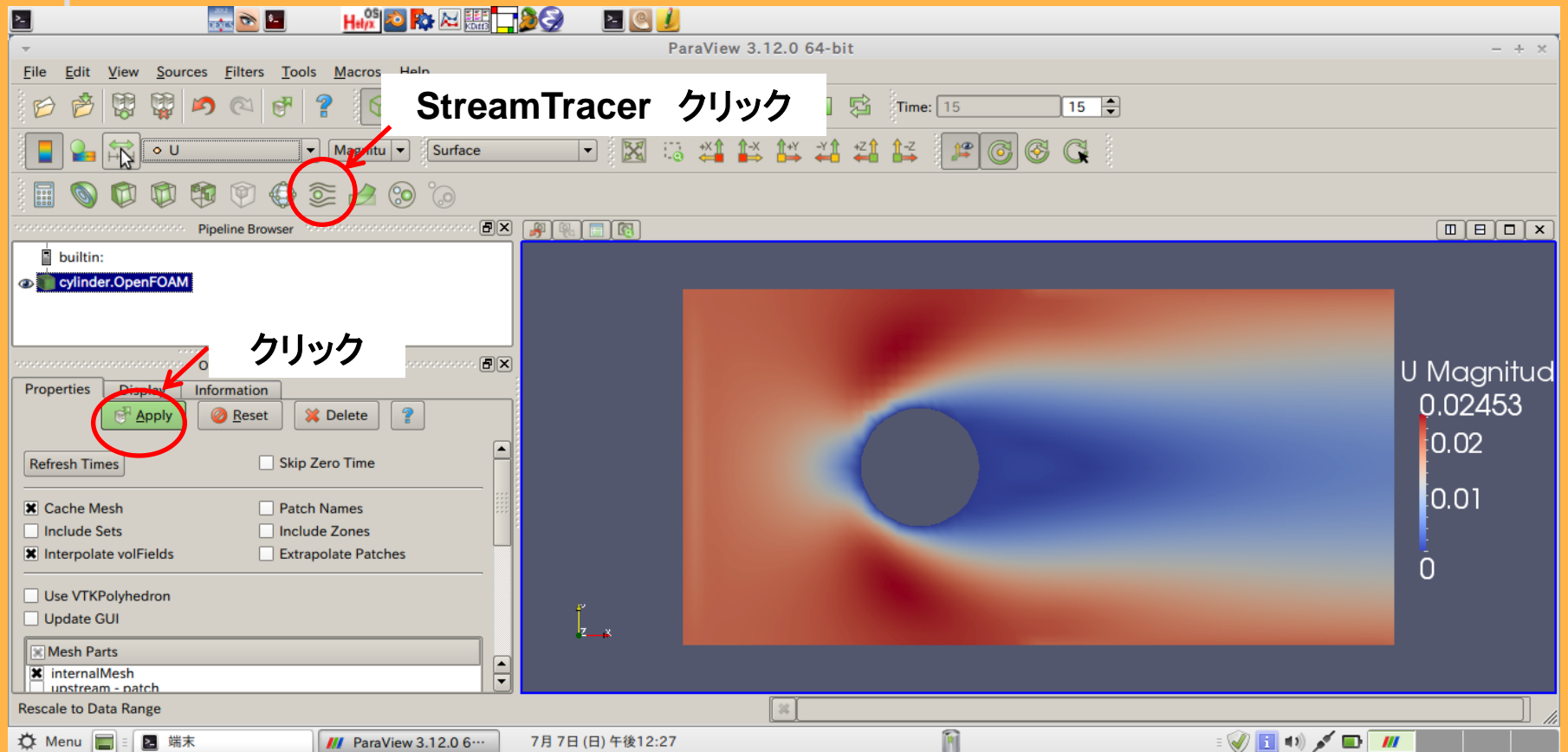
10秒程度で収束

paraFoam

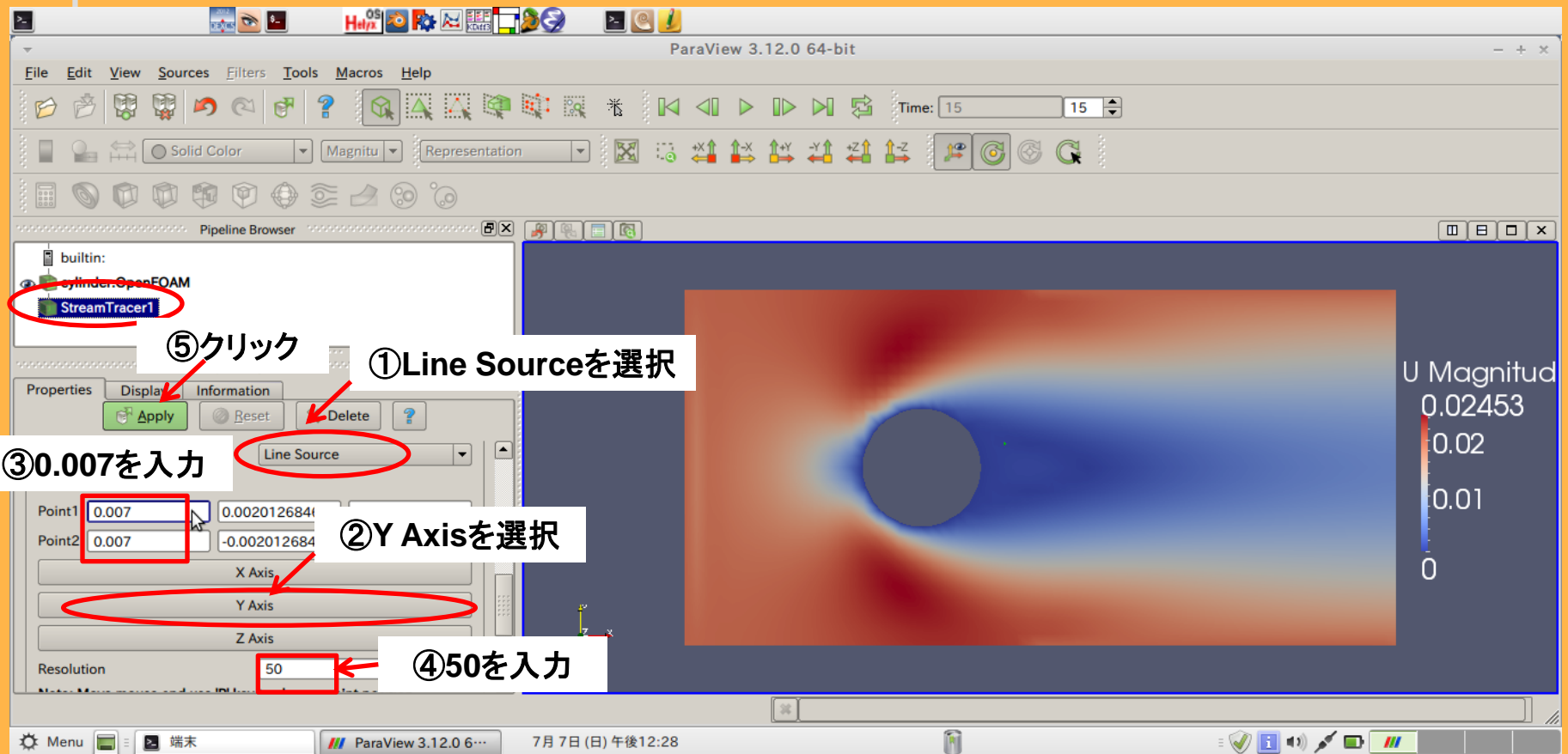


図の拡大⇒マウス中軸を回す

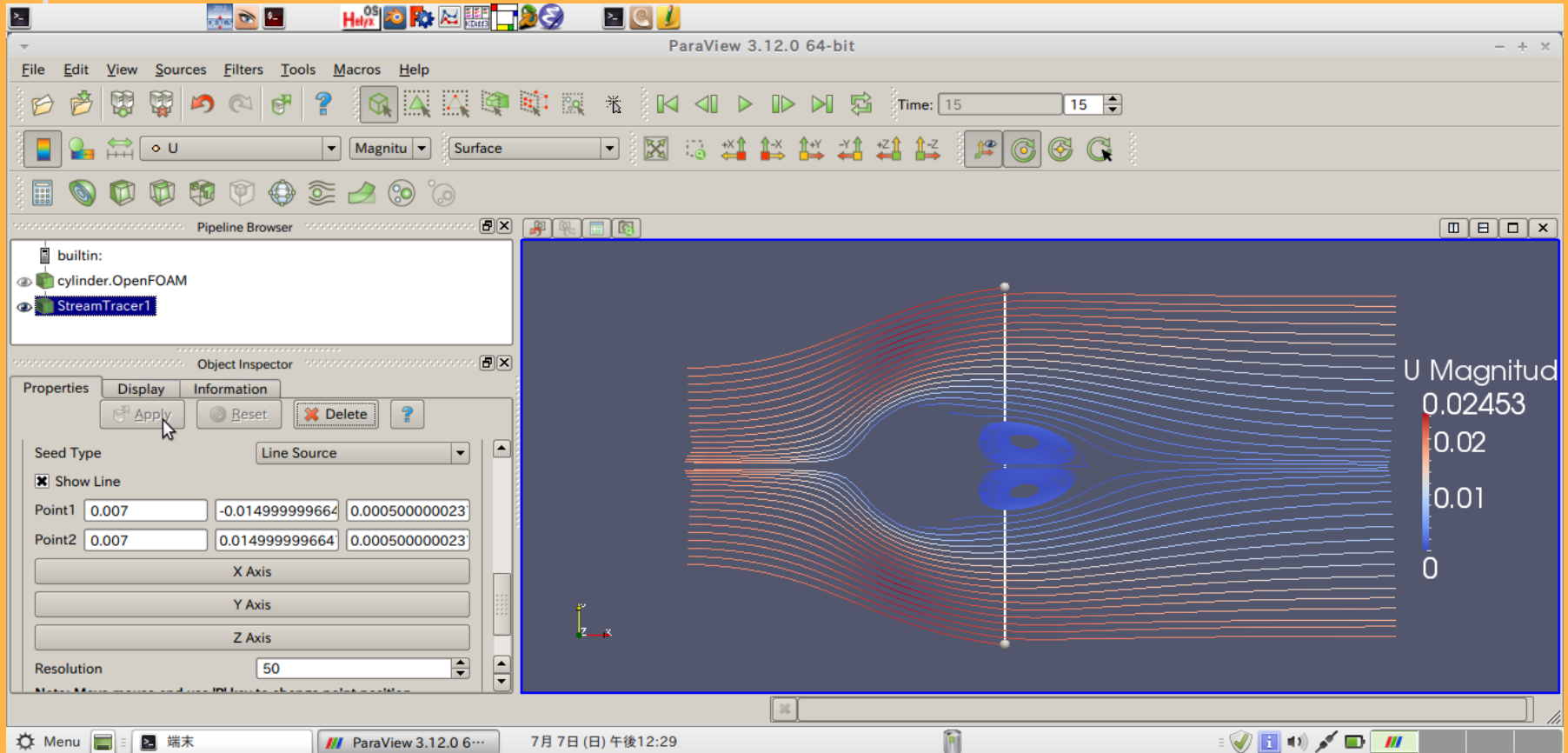
流線作図



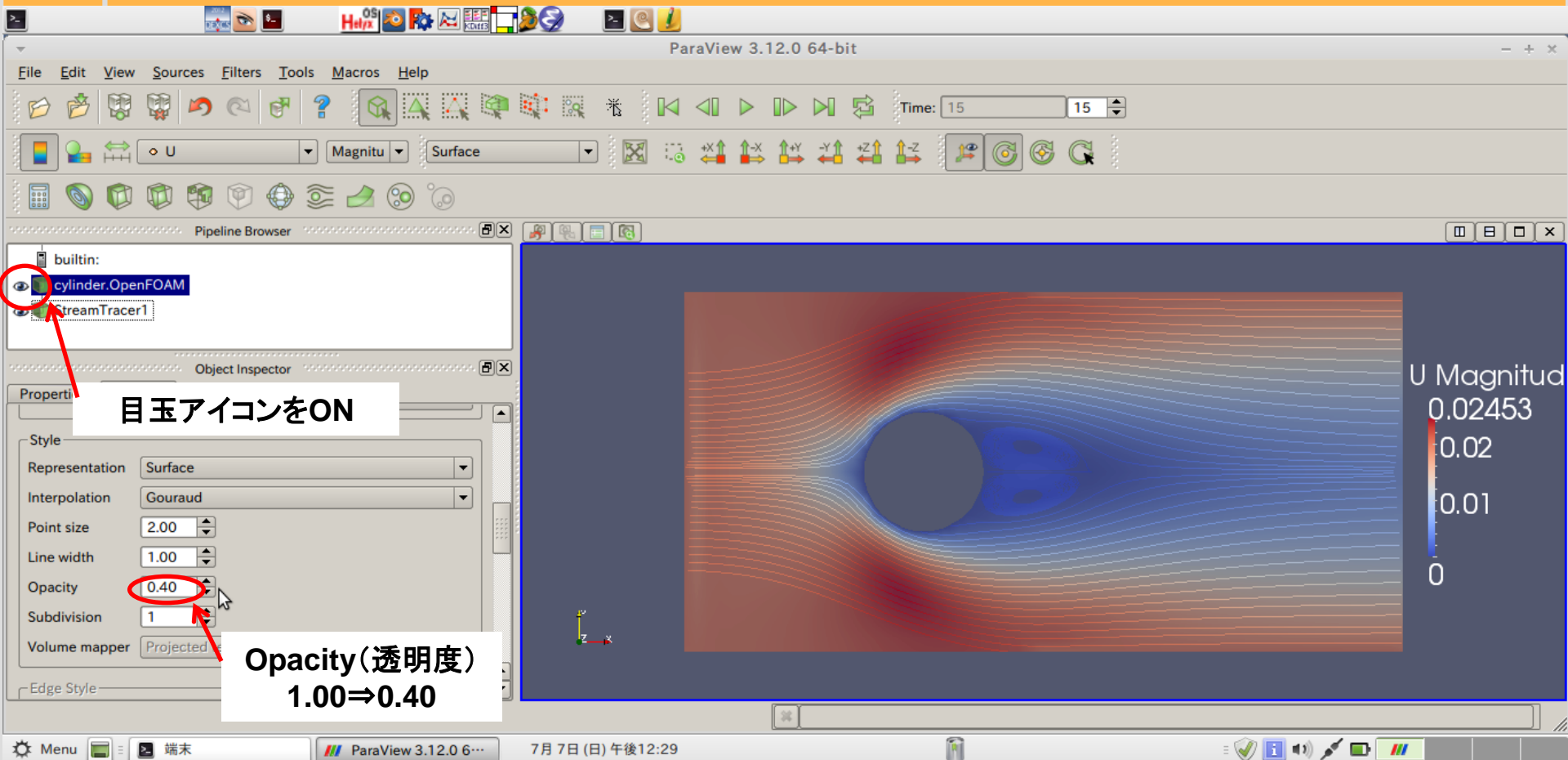
流線作図



流線作図



速度分布と流線の合図



まとめ

- pisoFoamを用いて2次元円柱周り流れの解析の演習を実施
- 解析の流れは以下のとおり。
 - ①円柱STLファイル作成 (FreeCAD)
 - ②blockMeshファイル作成・実行
 - ③snappyHexMeshファイル作成・実行
 - ④各種ファイル(境界、物性、制御)の作成
 - ⑤pisoFoamの実行
 - ⑥paraFoamを使った流線図の作成

付録 I

境界条件、メッシュ影響

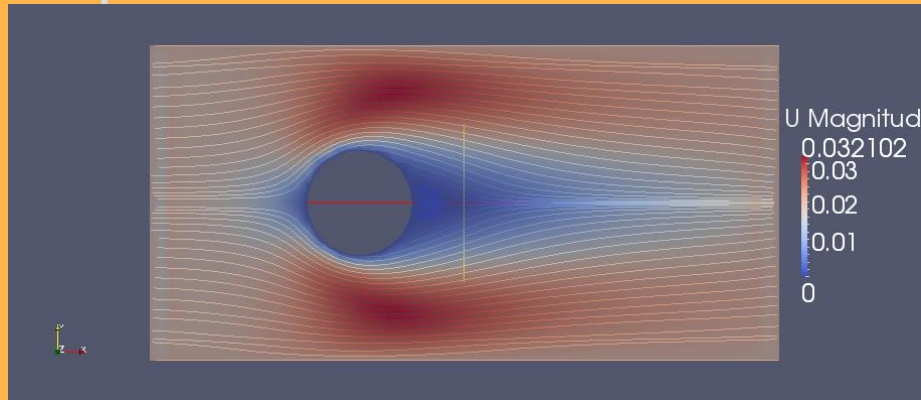
境界条件pの影響

Patch	U	p(case1)	p(case2)	p(case3)	p(case4)
Mesh	–	60 × 30 × 1	←	←	←
upstream	(2,0,0)	0	0	zeroGradient	zeroGradient
downstream	(2,0,0)	0	0	0	0
upANDdown	(2,0,0)	0	0	zeroGradient	zeroGradient
cylinder_zone0	(0,0,0)	zeroGradient	fixedValue; value \$internalField	fixedValue; value \$internalField	zeroGradient
frontANDback	empty	←	←	←	←
Cd	–	4.13	3.61	4.35	5.00
双子渦	–	△	×	×	△
計算時間	–	136	136	139	150

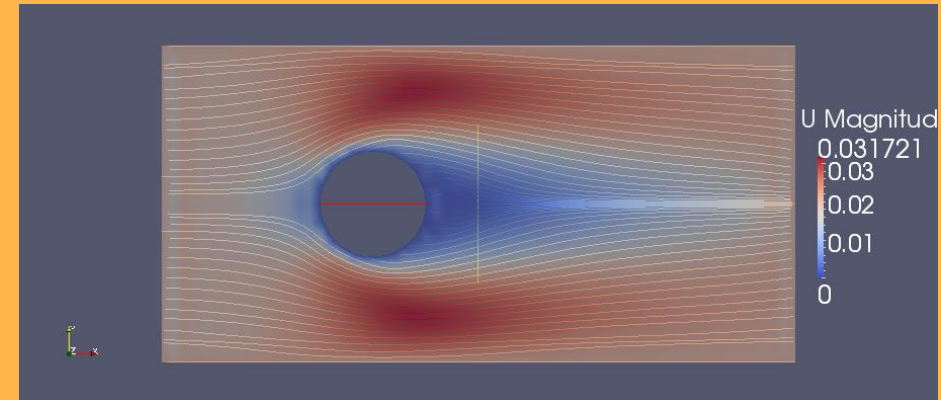
- ✓ Cd値は実験 (Cd=2.0) とかなり異なる
- ✓ 特に、円筒壁には \$internalField を使うのは不適

境界条件pの影響

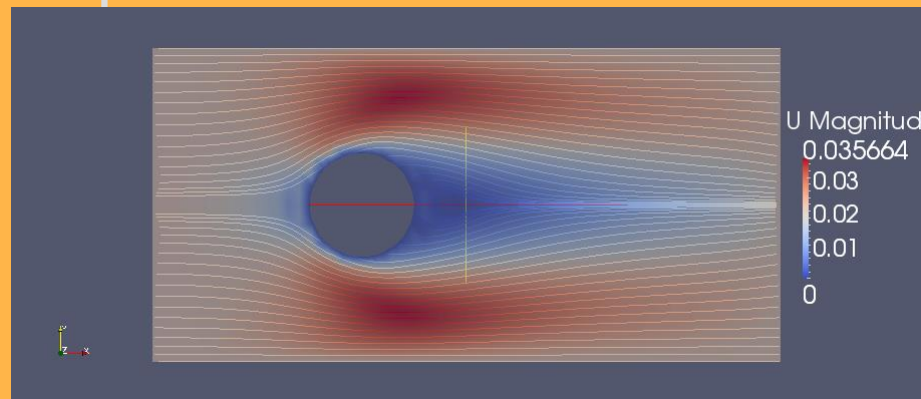
case1



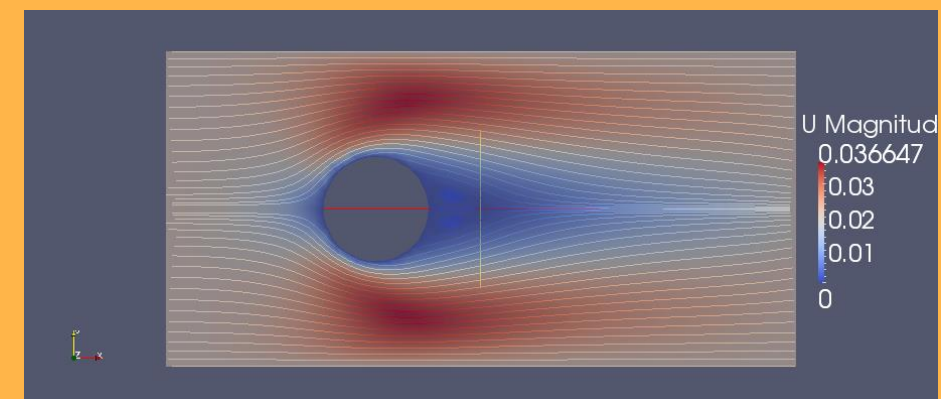
case2



case3



case4



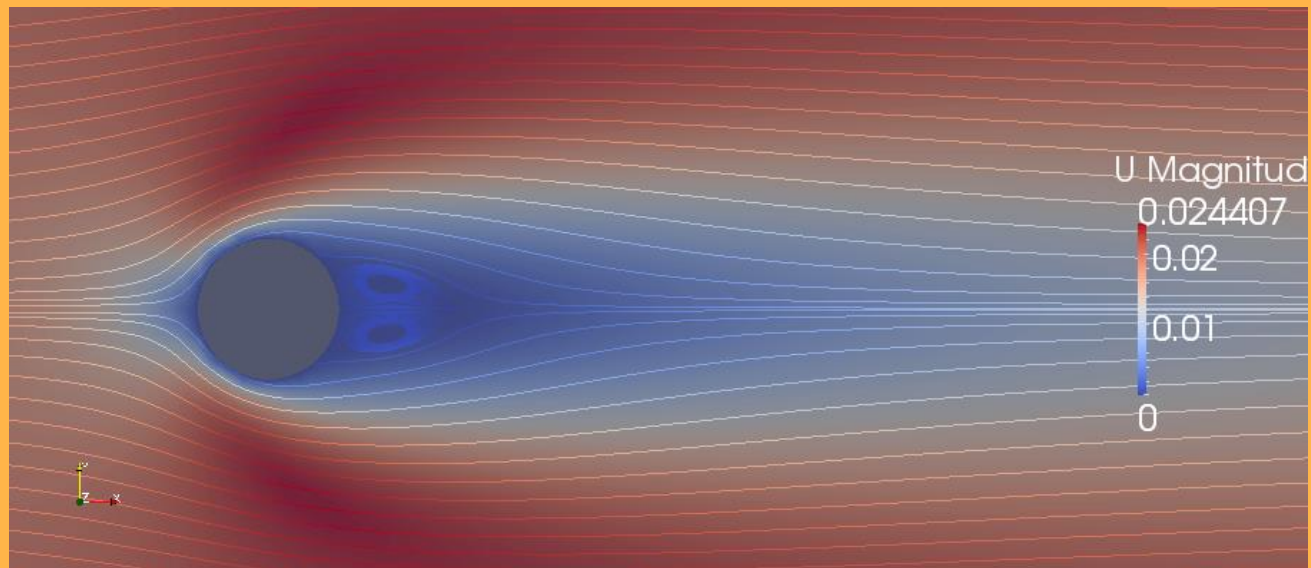
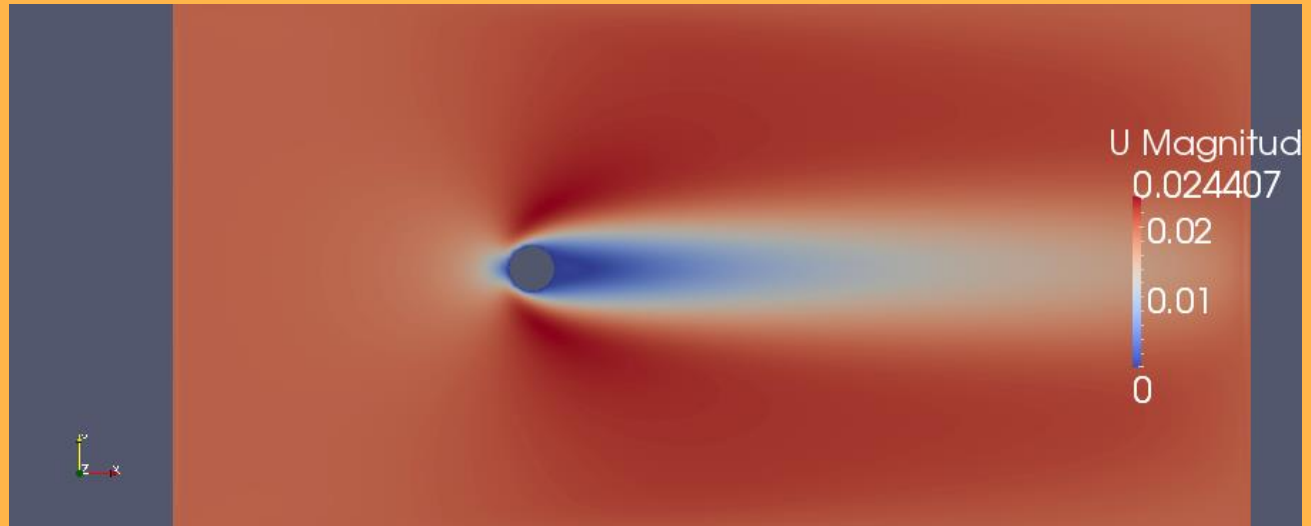
双子渦の様子が実験と異なる

計算領域の影響

Patch	U	p(case1)	p(case5)	p(case6)
$\Delta x - \Delta y$	–	6–3cm	24–12cm	28–16cm
Base Mesh	–	60 × 30	240 × 120	280 × 160
upstream	(2,0,0)	0	←	←
downstream	(2,0,0)	0	←	←
upANDdown	(2,0,0)	0	←	←
cylinder_zone0	(0,0,0)	zeroGradient	←	←
frontANDback	empty	empty	←	←
Cd	–	4.13	2.26	2.20
双子渦	–	Δ	\bigcirc	\bigcirc
計算時間	–	136	1,404	2,054

- ✓ 4倍程度、領域が広くなるとCdが収束してくる
- ✓ 但し計算時間は10倍

計算領域の影響(case5)



付録Ⅱ

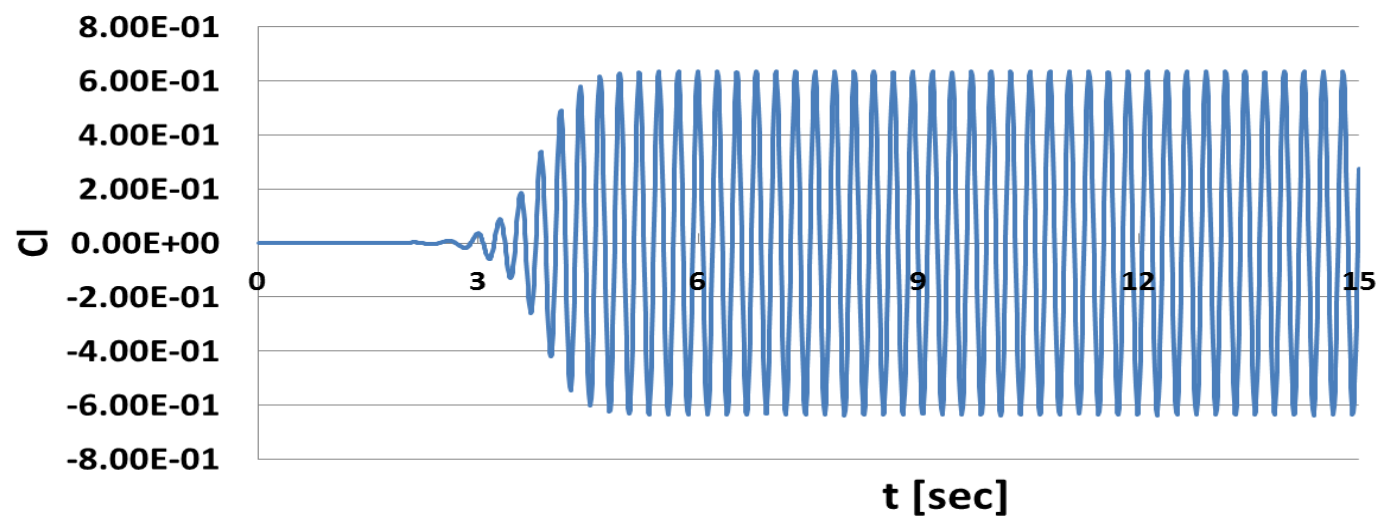
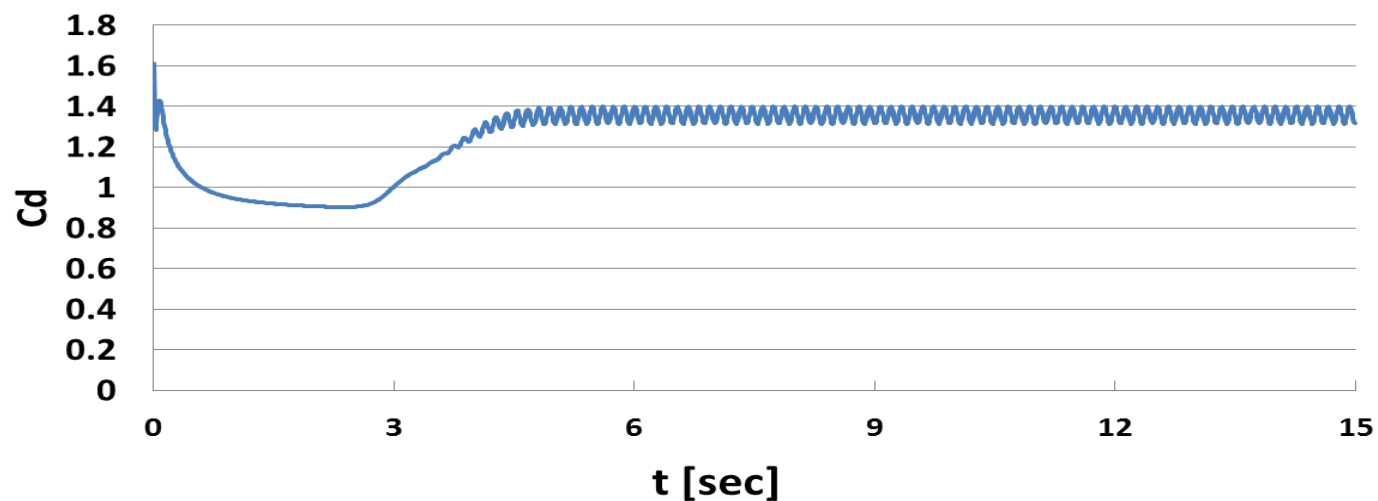
Re=200の場合

領域とメッシュの影響

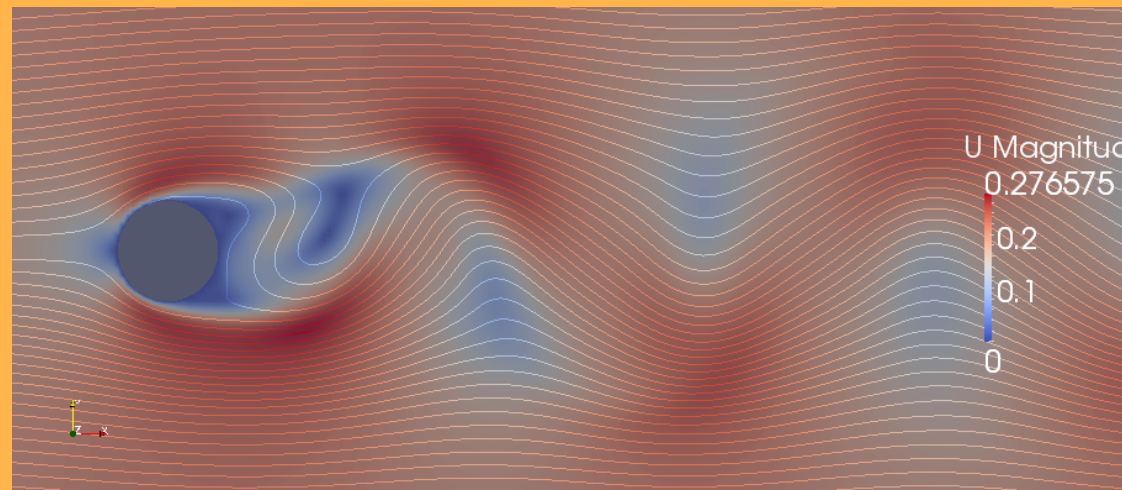
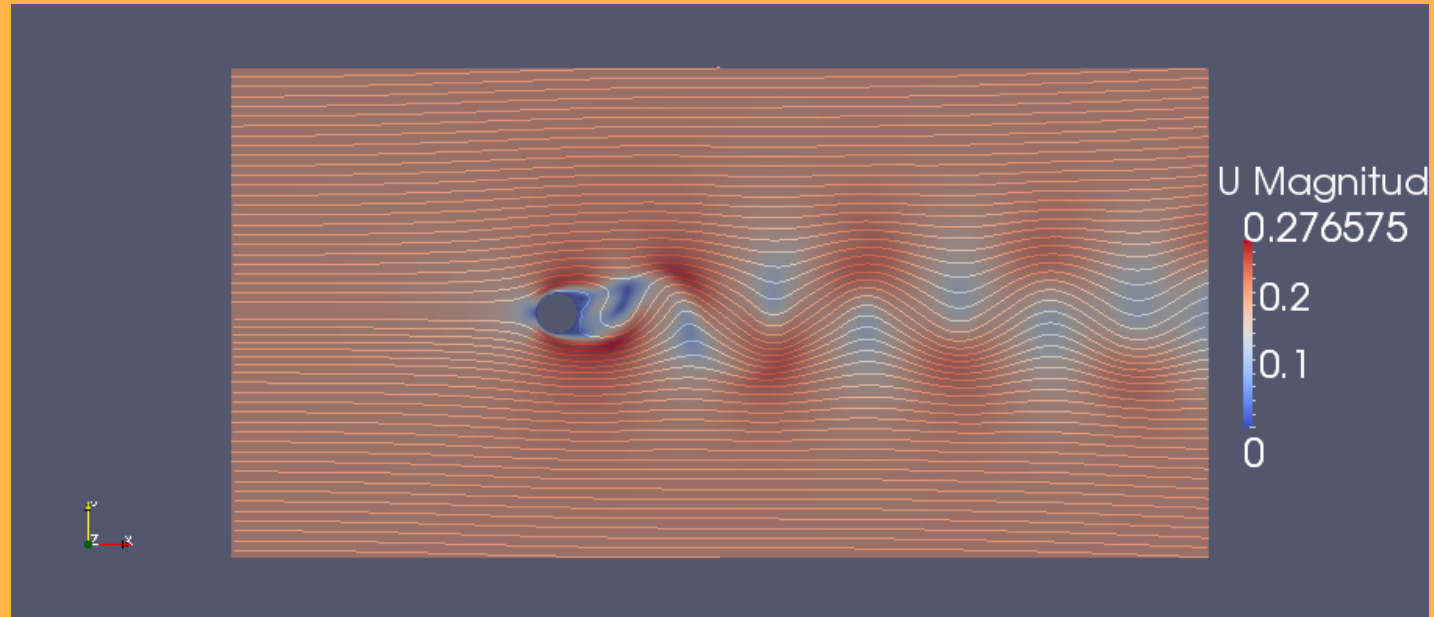
項目	case1	case2	case3
$\Delta x - \Delta y$	6-3cm	24-12cm	24-12cm
Base Mesh Cells	60 × 30	240 × 120	480 × 240 114,884
Cd	1.06	1.19	1.36
St	0.195	0.186	0.186
計算時間(sec)	129 (2.2min)	2,729 (45.5min)	19,987 (5.5hrs)

実験ではCd=1.5、St=0.19なのでCase3では程々の精度

CdとClの履歴(case3)



流れの様子 (case3)



付録Ⅲ

extrudeMeshを使った 2Dメッシュの作成

概要

- オープンCAE勉強会@富山/20130525第10回での富山大学の中川先生の資料を参考

http://eddy.pu-toyama.ac.jp/オープンCAE勉強会-富山/#_99

- snappyHexMeshで3次元メッシュを作成。extrudeMeshでPatch表面メッシュだけを押し出して2次元メッシュとする。
- 演習での方法では円柱周りメッシュが1層にできない場合があるが、本方法では必ず1層にできる。
- 本方法で領域のメッシュを部分的に詳細にすることが可能で計算時間の短縮になる。

blockMeshDictの修正

```
C blockMeshDict ✕
40 boundary
41 (
42     upstream
43     {
44         type patch;
45         faces
46         (
47             (0 4 7 3)
48         );
49     }
50     downstream
51     {
52         type patch;
53         faces
54         (
55             (1 2 6 5)
56         );
57     }
58     upANDdown
59     {
60         type patch;
61         faces
62         (
63             (0 1 5 4)
64             (3 7 6 2)
65         );
66     }
67     frontANDback
68     {
69         type empty;
70         faces
71         (
72             (0 3 2 1)
73             (4 5 6 7)
74         );
75     }
76 );
77
```

分解

```
57     }
58     upANDdown
59     {
60         type patch;
61         faces
62         (
63             (0 1 5 4)
64             (3 7 6 2)
65         );
66     }
67     front
68     {
69         type empty;
70         faces
71         (
72             (0 3 2 1)
73         );
74     }
75     back
76     {
77         type empty;
78         faces
79         (
80             (4 5 6 7)
81         );
82     }
83
```

snappyHexMeshDictの修正1

```
29 geometry
30 {
31     cylinder-s.stl
32     {
33         type triSurfaceMesh;
34         name cylinder;
35     }
36     // Analytical shape; cylinder, sphere
37
38     refinementBox1 領域名称
39     {
40         type searchableBox;
41         min ( -0.02 -0.03 0);
42         max ( 0.16 0.03 0.001);
43     }
44
45
46     refinementBox2 領域名称
47     {
48         type searchableBox;
49         min ( -0.01 -0.007 0);
50         max ( 0.06 0.007 0.001);
51     }
52 };
53
```

＜詳細メッシュ領域の指定＞
2つのボックスを指定
refinementBox1
refinementBox2
(名前は任意)

指定図形タイプ:ボックス

ボックスを規定できる2点を指定

指定図形タイプ:ボックス

ボックスを規定できる2点を指定

snappyHexMeshDictの修正2

＜詳細メッシュ領域の分割法指定＞

```
146 refinementRegions
147 { 領域名称
148     refinementBox1
149     {
150         mode inside;
151         levels ((1E15 1));
152     }
153     領域名称
154     refinementBox2
155     {
156         mode inside;
157         levels ((1E15 2));
158     }
159 }
```

ボックス内部／外部を指定

細分化レベル levels(距離、レベル)

ボックス内部／外部を指定

細分化レベル levels(距離、レベル)

refinementBox1の細分化レベルが1
に対しrefinementBox2はレベル2
でより細かく細分化される

extrudeMeshDictの作成

- pimpleDyMFoam/wingMotion/wingMotion2D_simpleFoam/system/**extrudeMeshDict**を/systemにコピー

```
10
17 // What to extrude:
18 //     patch    : from patch of another case ('sourceCase')
19
20 constructFrom patch;
21 sourceCase " ";
22 sourcePatches (front);
23
24 // If construct from patch: patch to use for back (can be same as sourcePatch)
25 exposedPatchName back;
26
27 // Flip surface normals before usage.
28 flipNormals false;
29
30 //- Linear extrusion in point-normal direction
31 extrudeModel linearNormal;
32
33 nLayers 1;
34
35 expansionRatio 1.0;
36
37 linearNormalCoeffs
38 {
39     thickness 0.001;
40 }
41
42 // Do front and back need to be merged? Usually only makes sense for 360
43 // degree wedges.
44 mergeFaces false;
```

現在のディレクトリでソースケースを参照

押し出すPatch: front

押し出されたPatch: back

押し出す層数

Uとpの修正

```
18 internalField      uniform (2.0E-1 0 0);
19 boundaryField
20 {
21     upstream
22     {
23         type          fixedValue;
24         value          uniform (2.0E-1 0 0);
25     }
26     downstream
27     {
28         type          inletOutlet;
29         inletValue     uniform (2.0E-1 0 0)
30         value          $internalField;
31     }
32     upANDdown
33     {
34         type          inletOutlet;
35         inletValue     uniform (2.0E-1 0 0)
36         value          $internalField;
37     }
38     front
39     {
40         type          empty;
41     }
42     back
43     {
44         type          empty;
45     }
46     cylinder_zone0
47     {
48         type          fixedValue;
49         value          uniform (0 0 0);
```

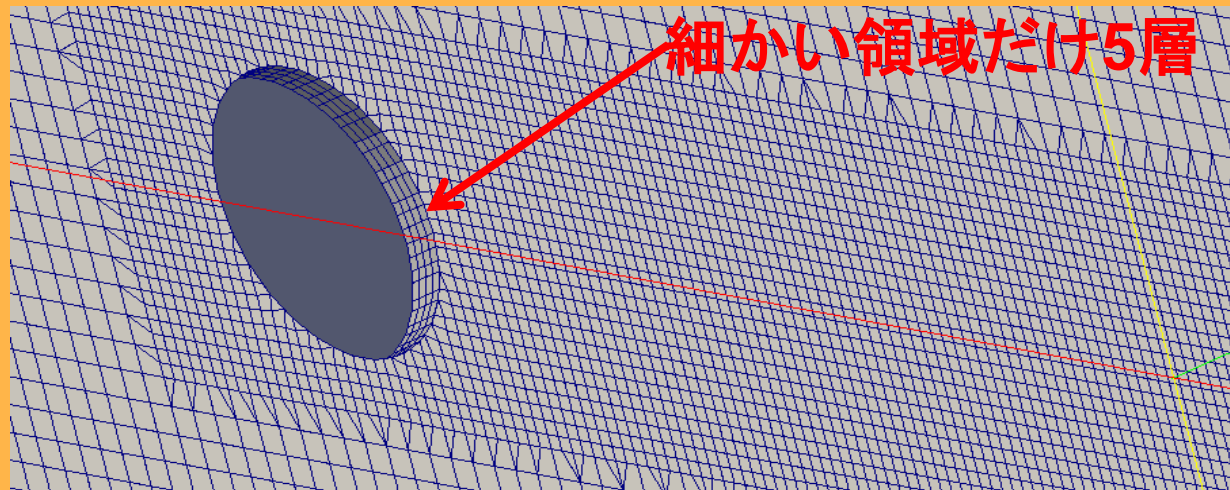
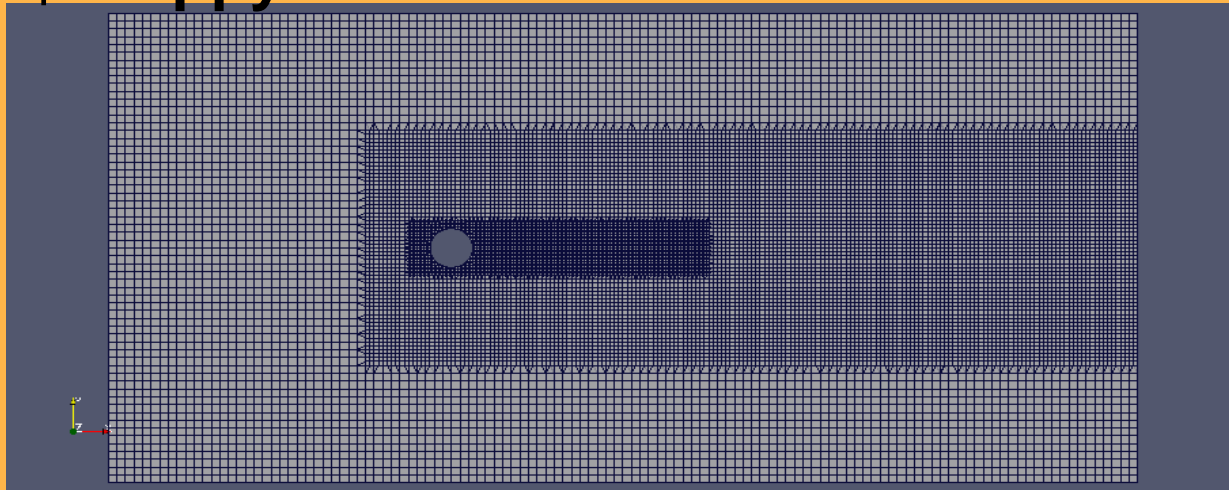
frontANDback
の分離

```
17 dimensions        [0 2 -2 0 0 0 0];
18 internalField      uniform 0;
19 boundaryField
20 {
21     upstream
22     {
23         type          fixedValue;
24         value          uniform 0;
25     }
26     downstream
27     {
28         type          fixedValue;
29         value          uniform 0;
30     }
31     upANDdown
32     {
33         type          fixedValue;
34         value          uniform 0;
35     }
36     front
37     {
38         type          empty;
39     }
40     back
41     {
42         type          empty;
43     }
44     cylinder_zone0
45     {
46         type          zeroGradient;
```

snappyHexMesh後のメッシュ

\$blockMesh

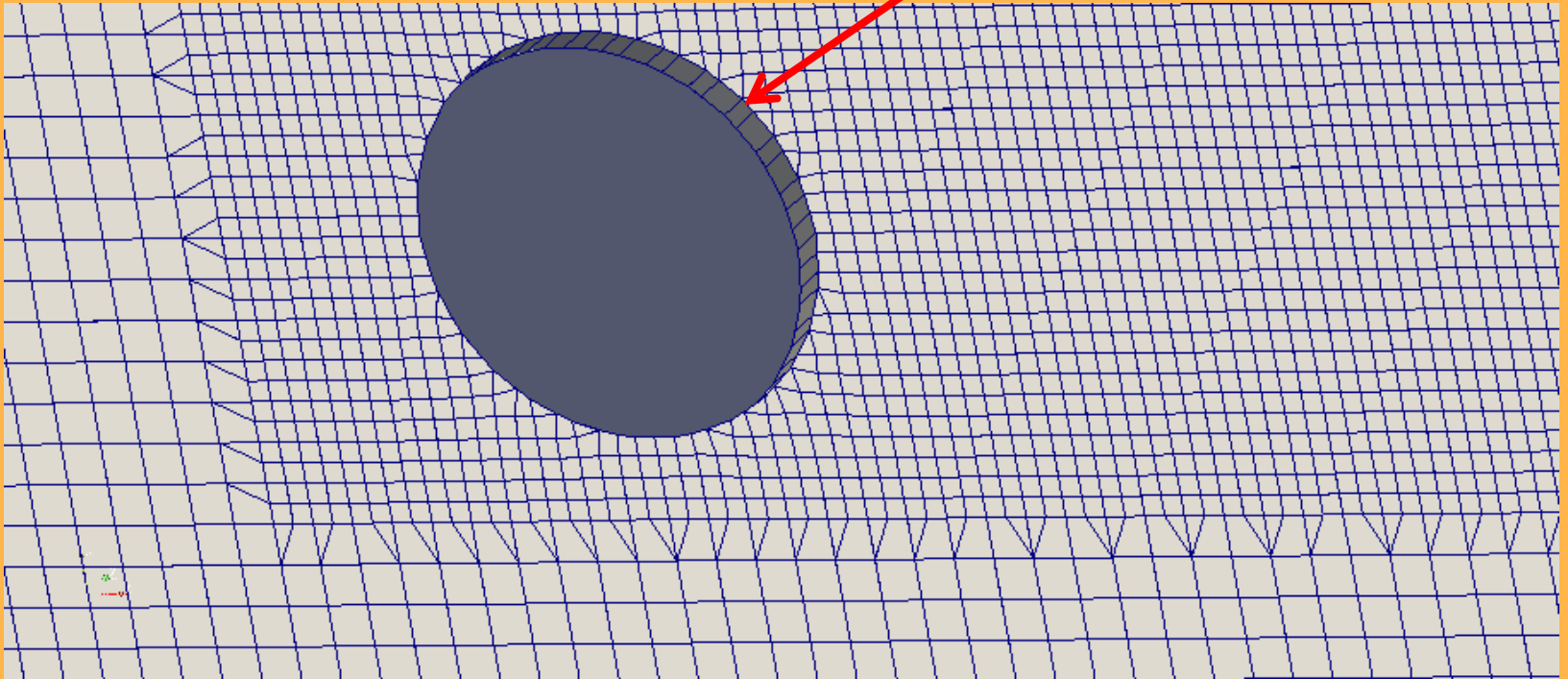
\$snappyHexMesh -overwrite



extrudeMeshの実行後のメッシュ

\$extrudeMesh

一層だけ

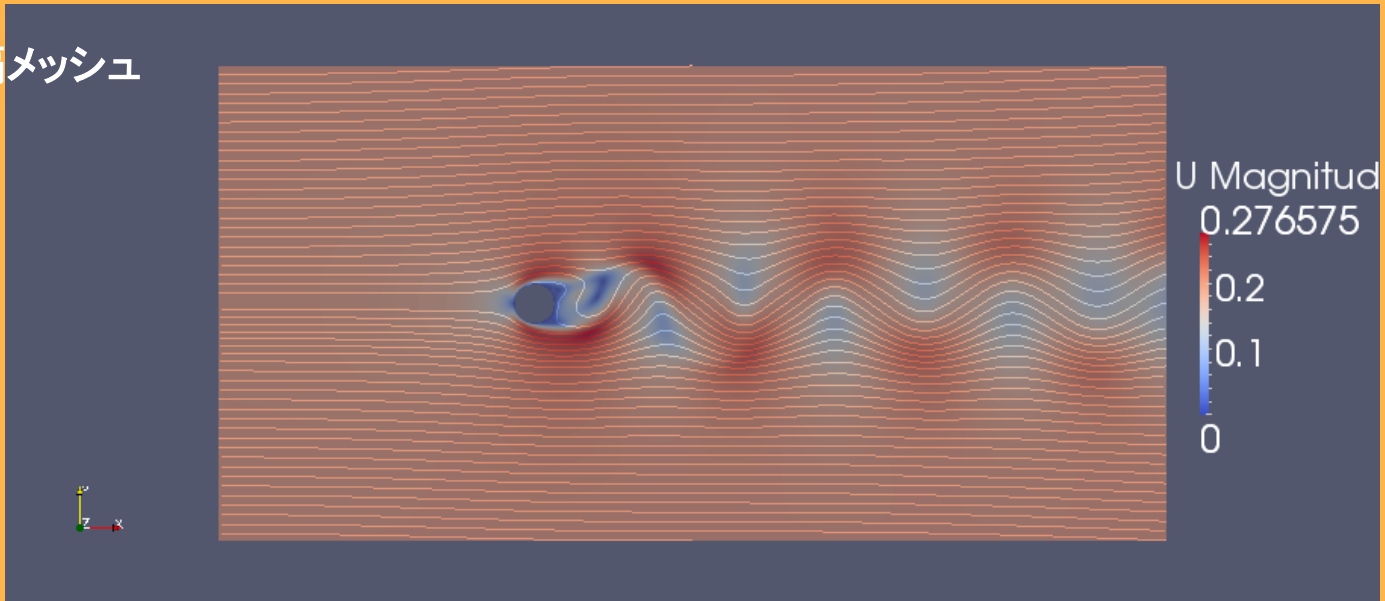


CdとStの比較

項目	今回	等間隔メッシュ
$\Delta x - \Delta y$	24-12cm	24-12cm
Cells	38,556	114,884
Cd	1.34	1.36
St	0.195	0.186
計算時間(sec)	1,880 (31.3min)	19,987 (5.5hrs)

流線

等間隔メッシュ



今回メッシュ

