

OPENFOAM(R) is a registered trade mark of OpenCFD Limited, the producer of the OpenFOAM software and owner of the OPENFOAM(R) and OpenCFD(R) trade marks.

平成22年度OpenFOAM®非圧縮性流体解析演習シリーズ

第4回

非等温流れ場hotRoomのチュートリアルを題材と
した吹出口、吸込口、発熱条件の設定

今野 雅 (オープンCAE学会、東京大学)



自己紹介

- 所属
 - 東京大学 大学院工学系研究科 建築学専攻
- 専門
 - 建築環境工学 (温熱・空気環境、特に数値予測)
- 所属学会
 - 日本建築学会
 - 空気調和・衛生工学会
 - 日本流体力学会
 - 日本風工学会
 - オープンCAE学会(副会長)



目次

1. 発熱面に対応する新たなパッチ作成と初期値の設定
2. 吹出口・吸込口に対応する新たなパッチ作成と初期値の設定
3. 質疑

端末の起動

The screenshot shows a Linux desktop environment with a dark theme. The application menu is open, displaying various categories like 'アプリケーション', 'アクセサリ', 'インターネット', etc. The 'アプリケーション' category is circled in red, and the '端末(OF-1.7.x)' application is also circled in red. A callout box points to the terminal window, which displays the following text:

```
端末
ファイル(F) 編集(E) 表示(V) 端末(T) ヘルプ(H)
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
dexcs@dexcs-desktop:~$
```

Two callout boxes provide instructions:

- 1. アプリケーション->DEXCS
- 2. 端末(OF-1.7.x)

A larger callout box at the bottom of the terminal window states:

端末が表われる
(OpenFOAM-1.7.x用
環境設定済み)

チュートリアル場所に行く

```
run ↵
```

```
cd tutorials/↵
```

```
cd heatTransfer/↵
```

```
cd buoyantBoussinesqSimpleFoam/↵
```

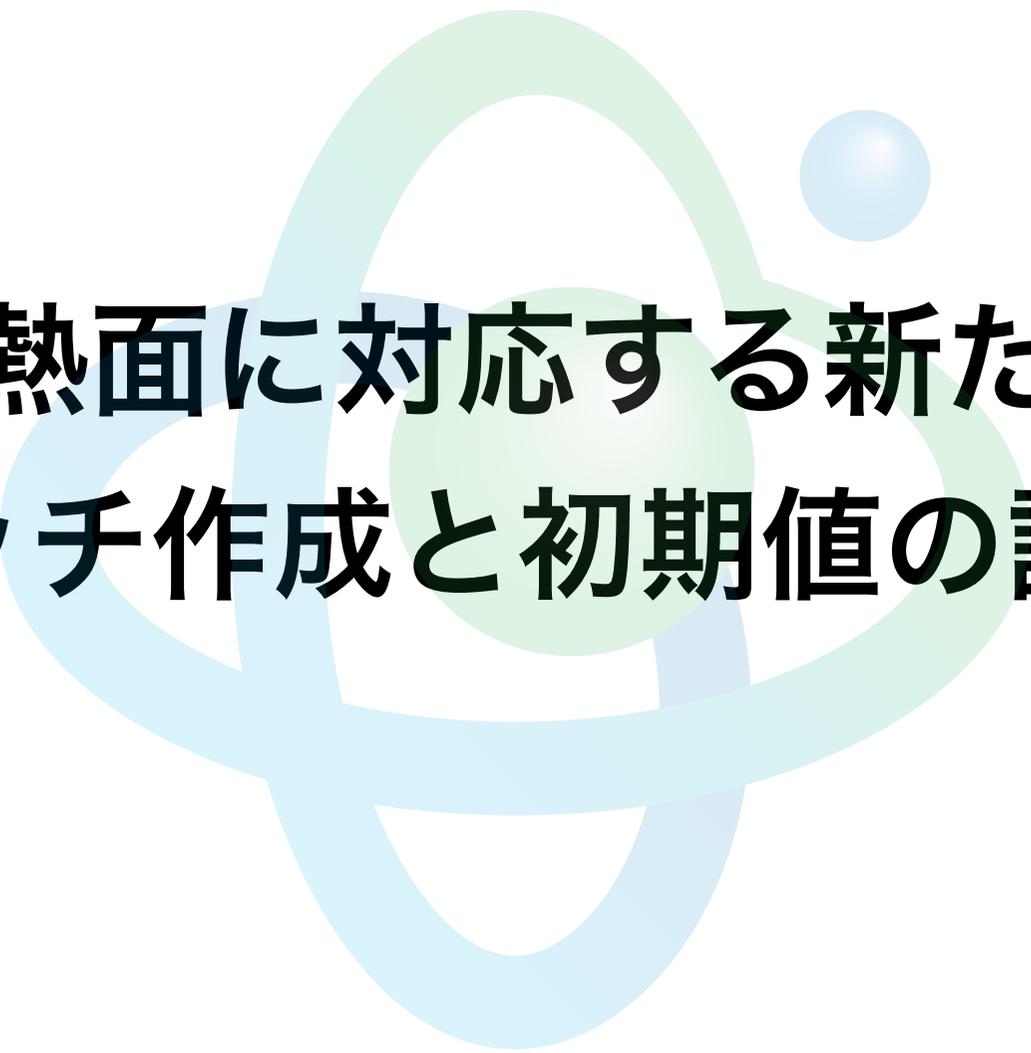
```
cd hotRoom/↵
```

```
ls ↵
```

←ディレクトリ名はTab
キーで補完できます

出力

```
0 Allclean Allrun constant system
```



発熱面に対応する新たな パッチ作成と初期値の設定

新たなパッチ作成と境界条件設定

1. ベース格子の生成と確認
2. faceSetユーティリティで境界面にする境界の集合(faceSet)を作成
3. faceSetからcreatePatchユーティリティを使い、新たなpatchを作成

新たなパッチ作成と境界条件設定

4. 新たなpatchに対する境界条件を、各物理量の初期ファイルで設定

5. 新たなパッチを含んだ新格子の確認

6. 計算の実行

実行結果の全消去

foamCleanTutorials ↵

←Tabキーで補完できます

出力

```
0 Allclean Allrun constant system
```

Allcleanがあるチュートリアル・ケースにおいては、
上記は ./Allclean を実行するのと同様です。

1. ベース格子生成と確認

blockMesh ↩

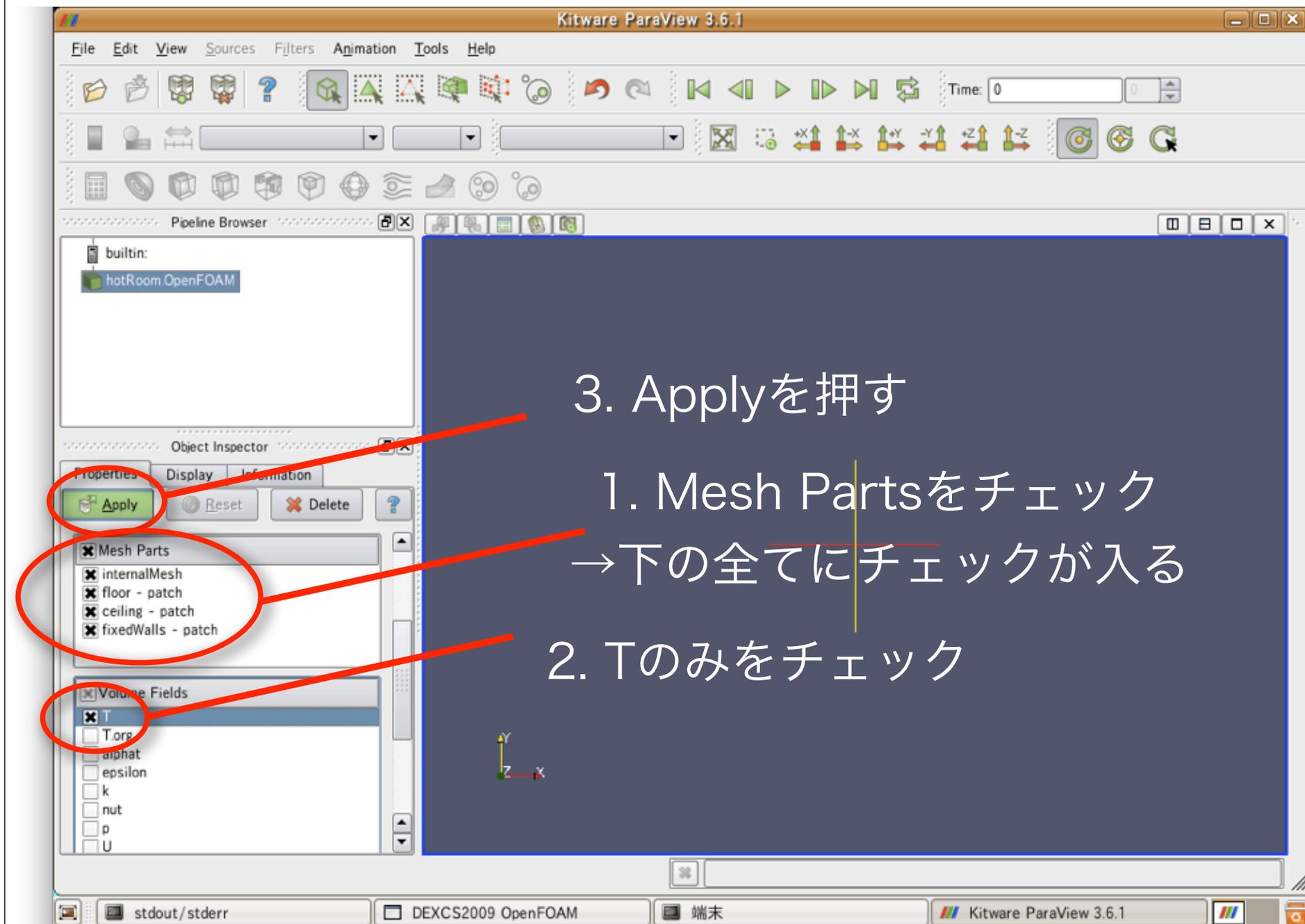
← ベースの格子生成

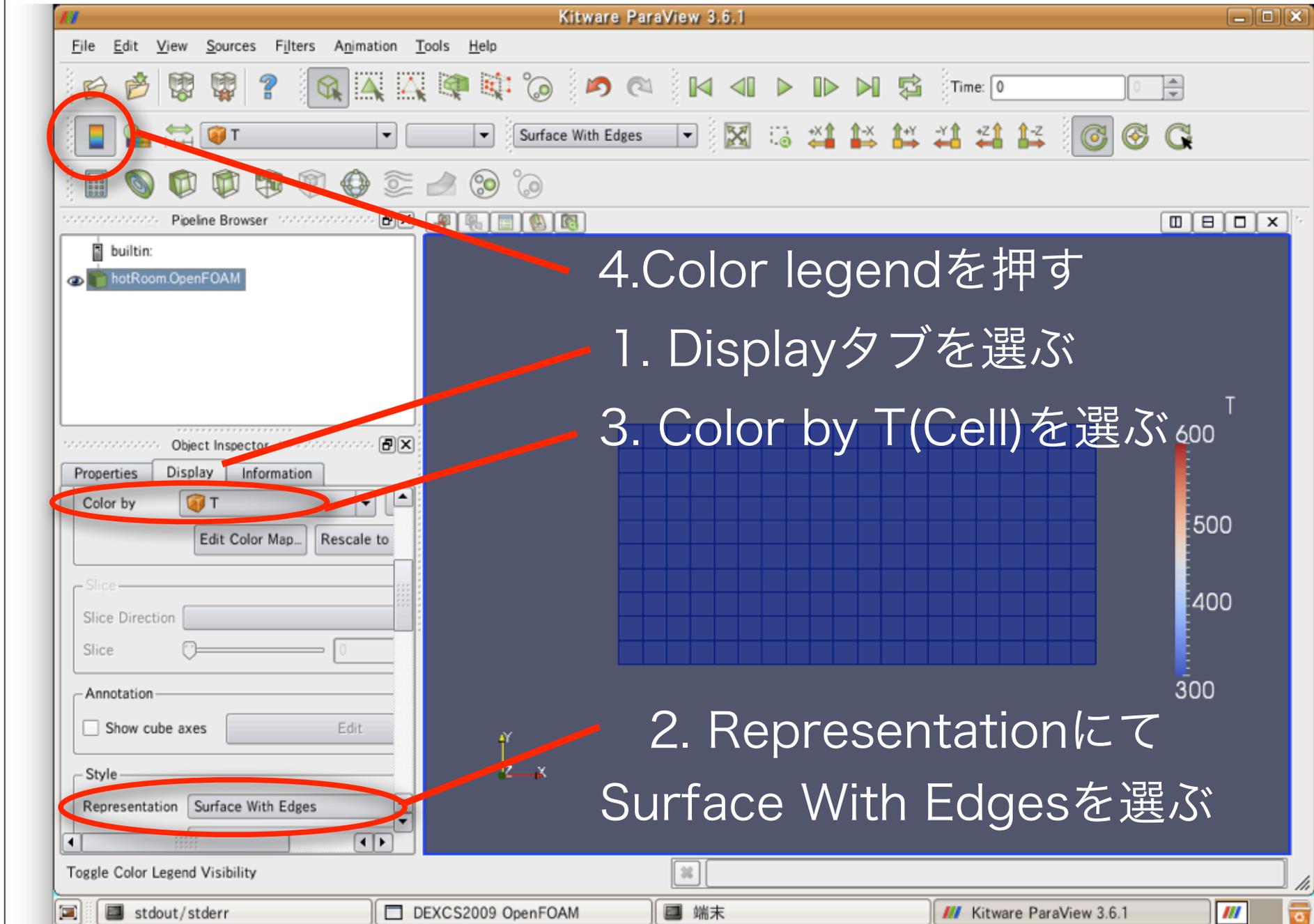
setHotRoom ↩

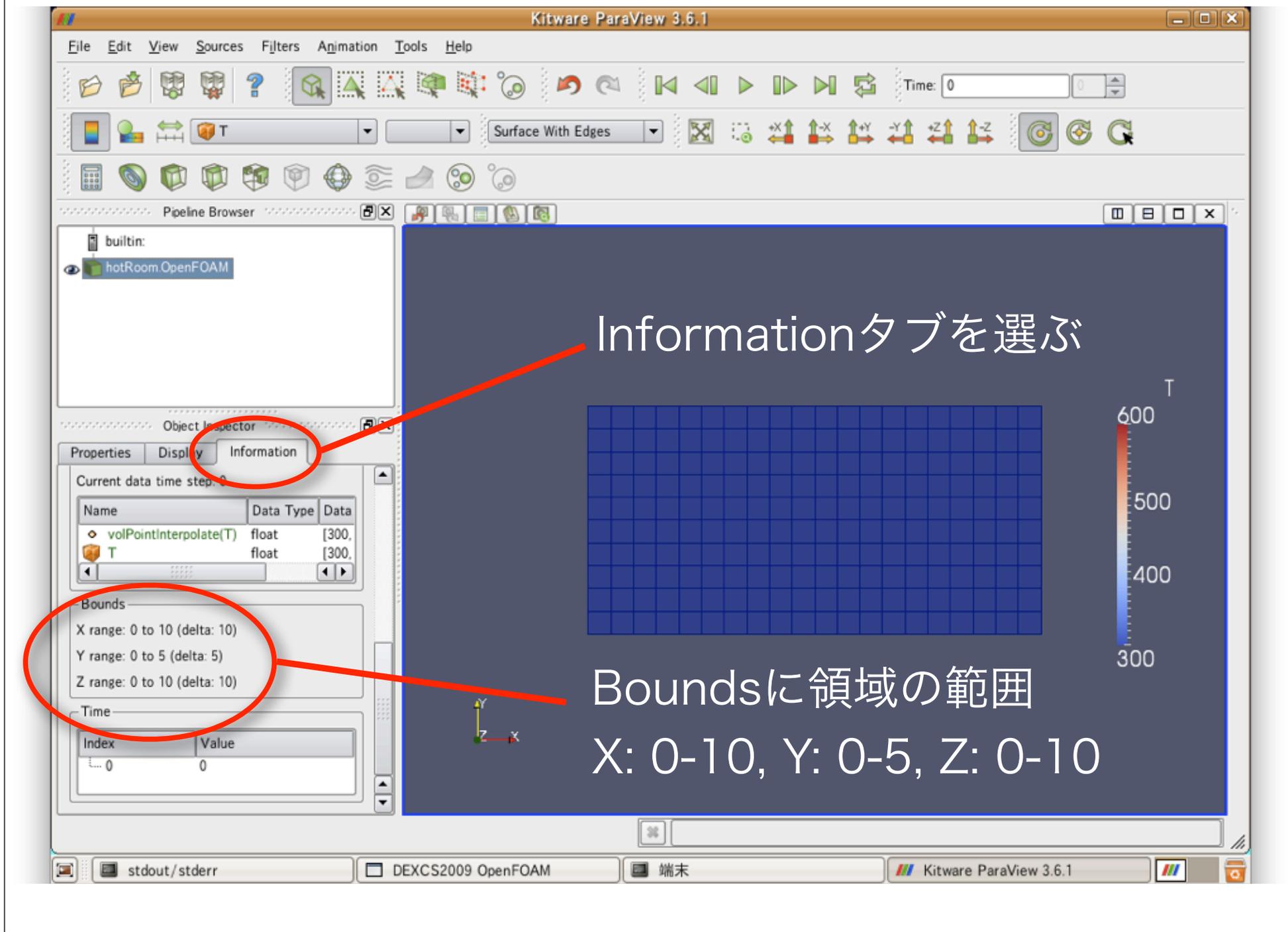
← 発熱面の設定 (専用Utility)

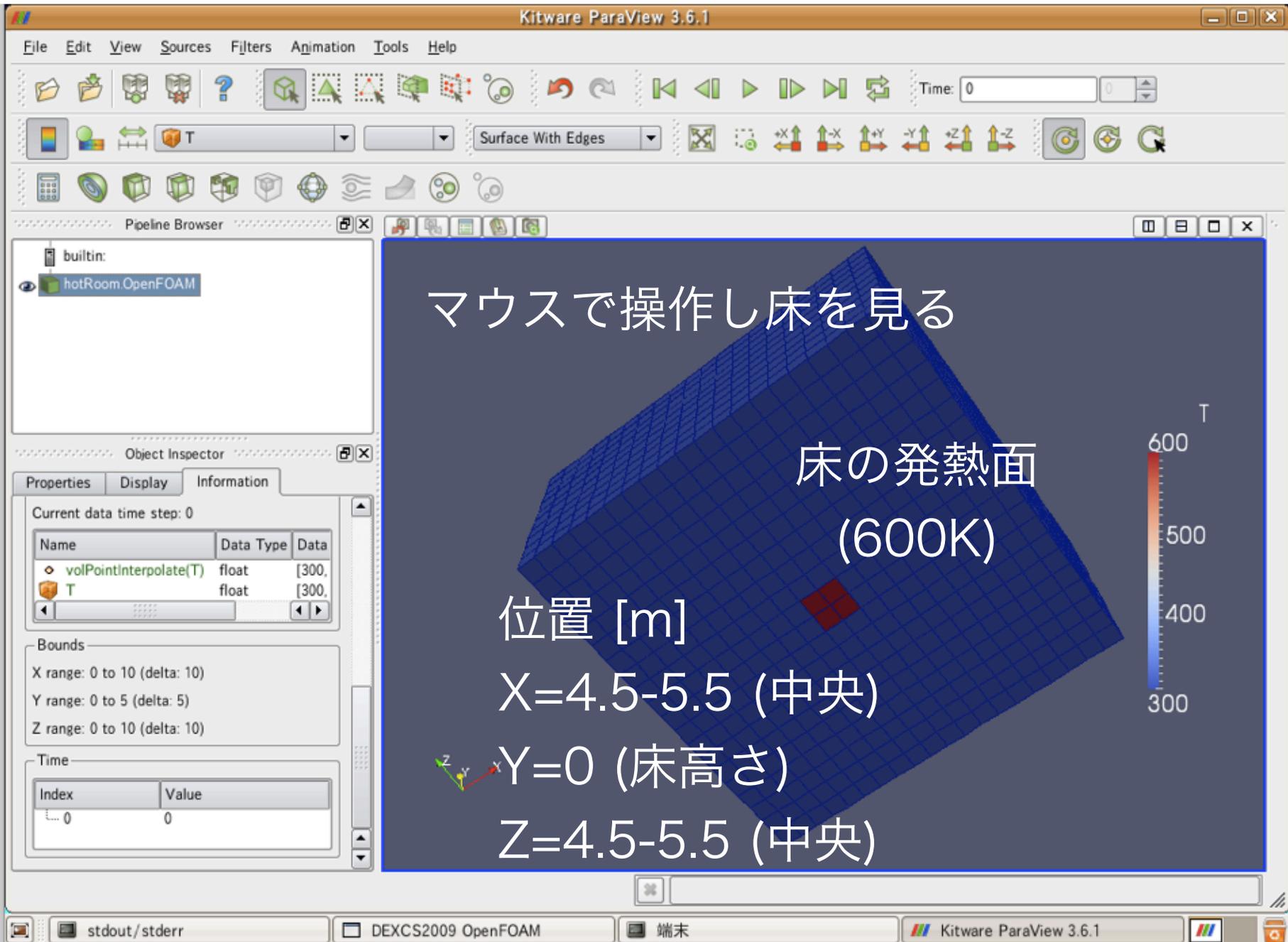
paraFoam ↩

← 可視化









発熱面の確認が終わったので、
初期化して格子のみ生成

foamCleanTutorials ↩

←ケースの初期化

blockMesh ↩

←格子生成



2. faceSetユーティリティで境界
面にする境界の集合(faceSet)を作成

faceSetDict例のコピー

```
cd system ↵
```

```
← *Dictはsystemに置く
```

```
find $FOAM_APP -name faceSetDict ↵
```

出力

```
/home/dexcs/OpenFOAM/OpenFOAM-1.7.x/applications/utilities/  
mesh/manipulation/faceSet/faceSetDict
```

1. アプリケーションの設定ファイルのサンプルは、そのソースのディレクトリーに置いてあることが多い。
2. FOAM_APPは、アプリケーションのソースがあるディレクトリーを示す環境変数。

faceSetDict例のコピー

↑ ↓ 前回のコマンドを ` (バッククォート) で囲む

```
cp `find $FOAM_APP -name faceSetDict` .
```

↓ (カレントディレクトリ) にコピー ↑

ls ↩

出力

```
controlDict faceSetDict fvSchemes fvSolution
```

` (バッククォート) で囲ったコマンドの出力が、外側のコマンドの引数になるので、"cp faceSetDictの場所 ." が実行され、faceSetDictがカレントディレクトリにコピーされる。

createPatchDict例のコピー

↑
cp `find \$FOAM_APP -name createPatchDict` . ↩
ls ↩

出力

```
controlDict  createPatchDict  faceSetDict  fvSchemes  
fvSolution
```

ファイラー等で手動で検索したり、コピーしたりする
のと異なり、シェルのコマンドで検索&コピー等の操
作行くと、似たような処理を繰り返す場合に便利。

各種設定ファイルの修正

```
gedit faceSetDict createPatchDict &↵
```

↑コマンド名やファイル名はTabキーで補完できます

1. geditで起動して、ファイル名を選んでも良いですが、最初から編集するファイル名を引数に指定すると早い。
2. もちろん、emacsやvi等、他の慣れているエディターを使っても構わない。
3. 最後に&を付けてバックグラウンドでgeditを動かす。

faceSetDictの修正

```
object faceSetDict;
}
// *****
// Name of set to operate on
name hotWall;
// One of clear/new/invert/add/delete|subset/list
action new;
// Actions to apply to pointSet. These are all the topoSetSource's ending
// in ..ToFace (see the meshTools library).
topoSetSources
(
  // Faces with face centre within box
  boxToFace
  {
    box (4.5 -0.001 4.5) (5.5 0.001 5.5);
  }
);
// ***
```

4. 「保存」を押す(Ctrl+Sでも可)

1.nameの値をhotWallに変更

2. boxToFaceのブロック
以外は消去

3. boxの領域を加熱壁を囲む領域に変更する
(Y=0の境界面を囲むのでY=- ϵ ~ ϵ 。 ϵ : 微小量)

faceSetの実行

画面下の端末のタブを押して、端末で入力

```
cd .. ↵
```

←ケースディレクトリに戻る

```
faceSet ↵
```

出力

```
Reading faceSetDict
```

```
Backing up hotWall into hotWall_old
```

```
Set:hotWall Size:0 Action:new
```

```
    Adding faces with centre within box (4.5 -0.001 4.5)  
(5.5 0.001 5.5)
```

```
Writing hotWall (size 4) to "constant/polyMesh/sets/hotWall"
```

4つの格子界面のfaceSetが、constant/polyMesh/
sets/hotWallとして書き出された。

hotWallのfaceSetの確認

[more constant/polyMesh/sets/hotWall](#) ↩

moreのキー/スペース：続ける、b：バック、q：終了

出力

```
4  
(  
11389  
11390  
11409  
11410  
)
```

faceSetの実際の中身は格子界面の番号のリスト
(通常、中身を詳しく見る必要はない。)



3. faceSetからcreatePatch ユーティリティを使い、新たな patchを作成

createPatchDictの修正

5. ファイル→終了 (Ctrl+Qでも可)

1. createPatchDictのタブ選択

3. patch名(name)の値をhotFloorに変更

2. constructFrom setのブロック以外は消去

4. setの値を先程作成したfaceSet名のhotWallに変更

```
// Patches to create.
patchInfo
(
  {
    name hotFloor;

    // Type of new patch
    dictionary
    {
      type wall;
    }

    constructFrom set;
    patches ();

    set hotWall;
  }
);
```

```
// ***** //
```

(54 行, 22 列)

[挿入]

createPatch

createPatch -overwrite ↩

↑ 通常は新格子が次の時刻ディレクトリに出力されるが、`overwrite`オプションで現格子を上書きする

出力

Adding new patch hotFloor as patch 3 from

{

type wall;

(省略)

Read 4 faces from faceSet hotWall

(省略)

Writing repatched mesh to 0

hotWallのfaceSetからなる新しいpatchである
hotFloorが、0(constant/polyMesh)に上書きで出力

格子の境界の確認

more constant/polyMesh/boundary ↩

スペース：続ける、b：バック、q：終了

出力

```
hotFloor
{
    type            wall;
    nFaces          4;
    startFace       12796;
}
```

格子の境界情報を表わすboundaryにhotFloorのpatchがtype wallとして追加されている。
(通常、中身を詳しく見る必要はない。)



4. 新たなpatchに対する境界条件
を、各物理量の初期ファイルで設定

各物理量の初期ファイルの修正

```
ls -al 0/
```

```
rm 0/p 0/alphat
```

```
gedit 0/* & ↵
```

v1.7で解くのは $p \rightarrow p_rgh$

alphat \rightarrow kappat

1. 各物理量の初期ファイルは通常0/の下
2. *(アスタリスク)は.(ドット)で始まるファイルやディレクトリ以外 (ワイルドカード) に置換され、全初期ファイルが編集対象となる。

温度の初期ファイル(0/T)の修正

1. Tのタブが選択されていることを確認

```
internalField    uniform 300;

boundaryField
{
    floor
    {
        type      fixedValue;
        value     uniform 300;
    }
    hotFloor
    {
        type      fixedValue;
        value     uniform 600;
    }
    ceiling
    {
        type      fixedValue;
        value     uniform 300;
    }
    fixedWalls
    {
        type      zeroGradient;
    }
}
```

2. floorのpatchのvalueはnonuniformから、uniform 300に変更。

3. hotFloorのpatchを追加

4. valueはuniform 600

(30行、36列) [挿入]

速度の初期ファイル(0/U)の修正

1. Uのタブ選択

```
internalField uniform (0 0 0);

boundaryField
{
  floor
  {
    type fixedValue;
    value uniform (0 0 0);
  }
  hotFloor
  {
    type fixedValue;
    value uniform (0 0 0);
  }
  ceiling
  {
    type fixedValue;
    value uniform (0 0 0);
  }
}
```

3. hotFloorのpatchを、floorのpatchからコピーして追加

(31行, 36列) [挿入]

その他の初期ファイルの修正

boundaryField
{
 floor
 {
 type alphasFunction;
 value uniform 0;
 }
 hotFloor
 {
 type alphasFunction;
 value uniform 0;
 }
 ceiling
 {
 }
 fixe
 {
 type alphasFunction;
 value uniform 0;
 }
}

T.org以外も同様に hotFloorのpatchを、floorのpatchからコピーして追加

(23行、2列) [挿入]

その他の初期ファイルの修正

The screenshot shows a gedit window with the following menu items: ファイル(F) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H). The toolbar includes icons for New, Open, Save, Print, Undo, Redo, Cut, Copy, Paste, Find, and Replace. The window title is `*p (~ / OpenFOAM / dexcs - 1.6.x / run / tutorials / heatTransfer / buoyantBoussinesqSimpleFoam / hotRoom / 0) - gedit`. The main text area contains code for boundary conditions:

```
boundaryField
{
    floor
    {
        type
        rho
        value
    }
    hotFloor
    {
        type
        rho
        value
    }
    ceiling
    {
        type
        rho
        value
    }
    uniform U;
```

The dialog box contains the following text:

まだ保存していないドキュメントが 7 個あります。閉じる前にそれらを保存しますか？

保存するドキュメントを選択して下さい(E):

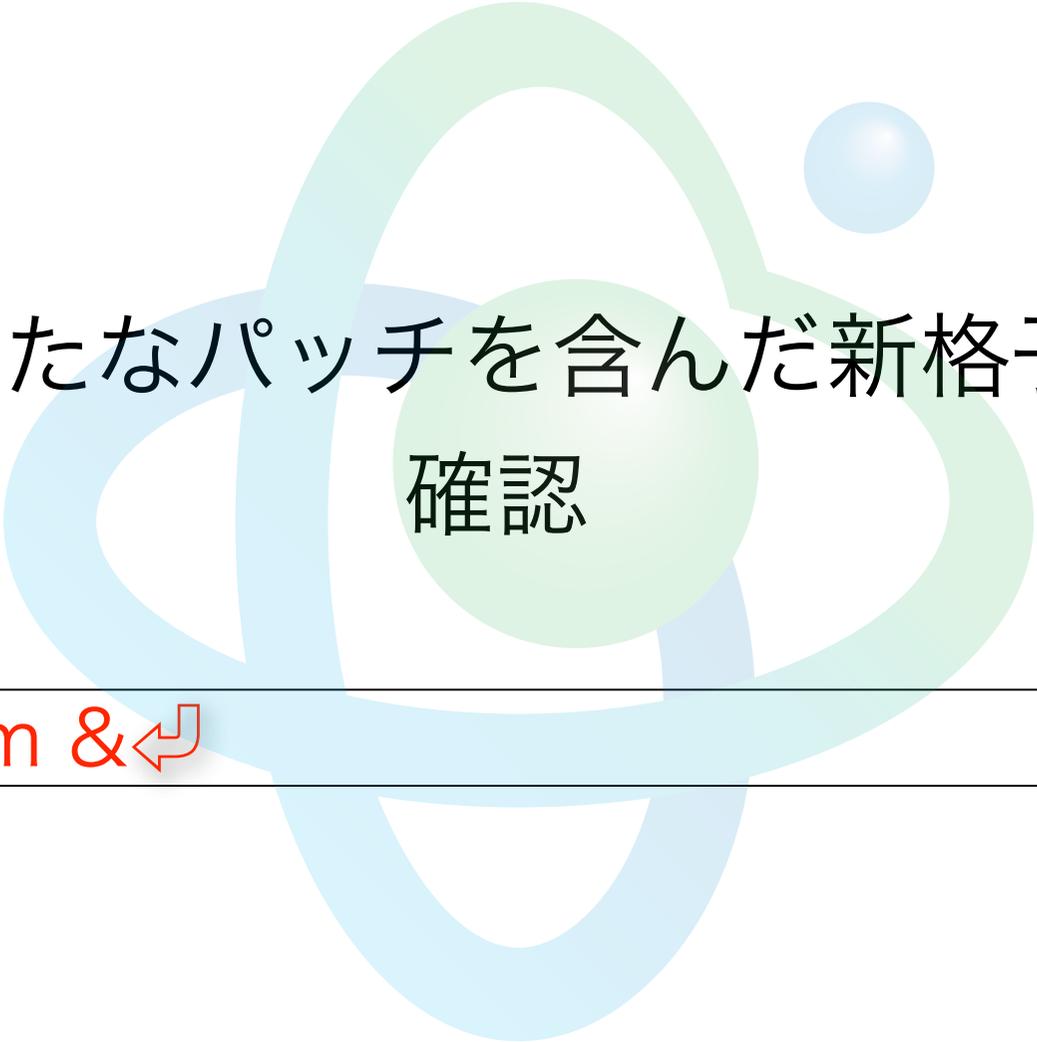
- T
- U
- alphas
- epsilon
- k
- ...

保存しないと変更した内容が完全に失われます。

Buttons: 保存せずに閉じる(W) キャンセル(C) **保存(S)**

Bottom text box: Ctrl+Qで終了し、変更を全て保存

Bottom right status bar: (24 行, 6 列) [挿入]



5. 新たなパッチを含んだ新格子の 確認

paraFoam & ↩

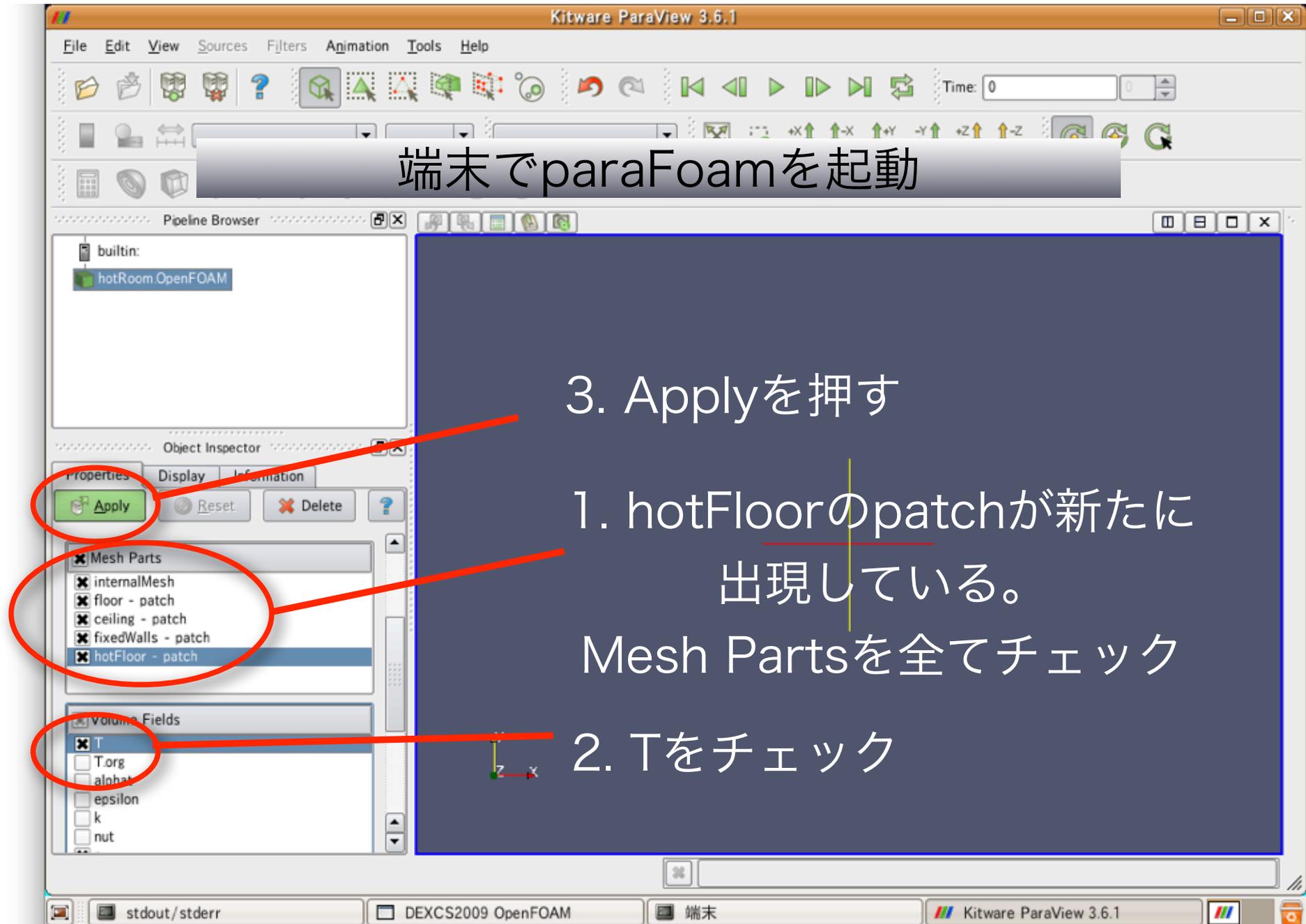
端末でparaFoamを起動

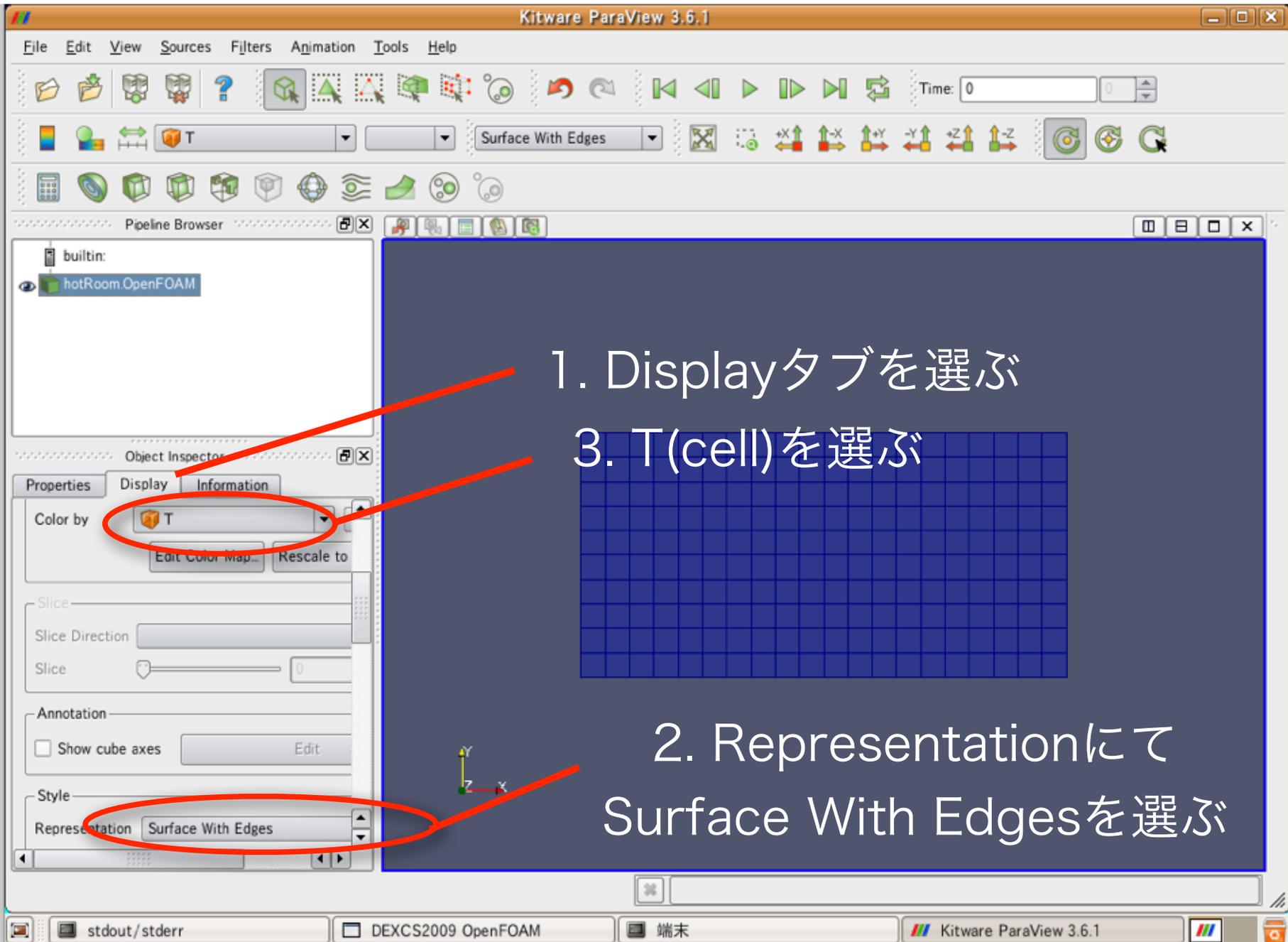
3. Applyを押す

1. hotFloorのpatchが新たに
出現している。

Mesh Partsを全てチェック

2. Tをチェック

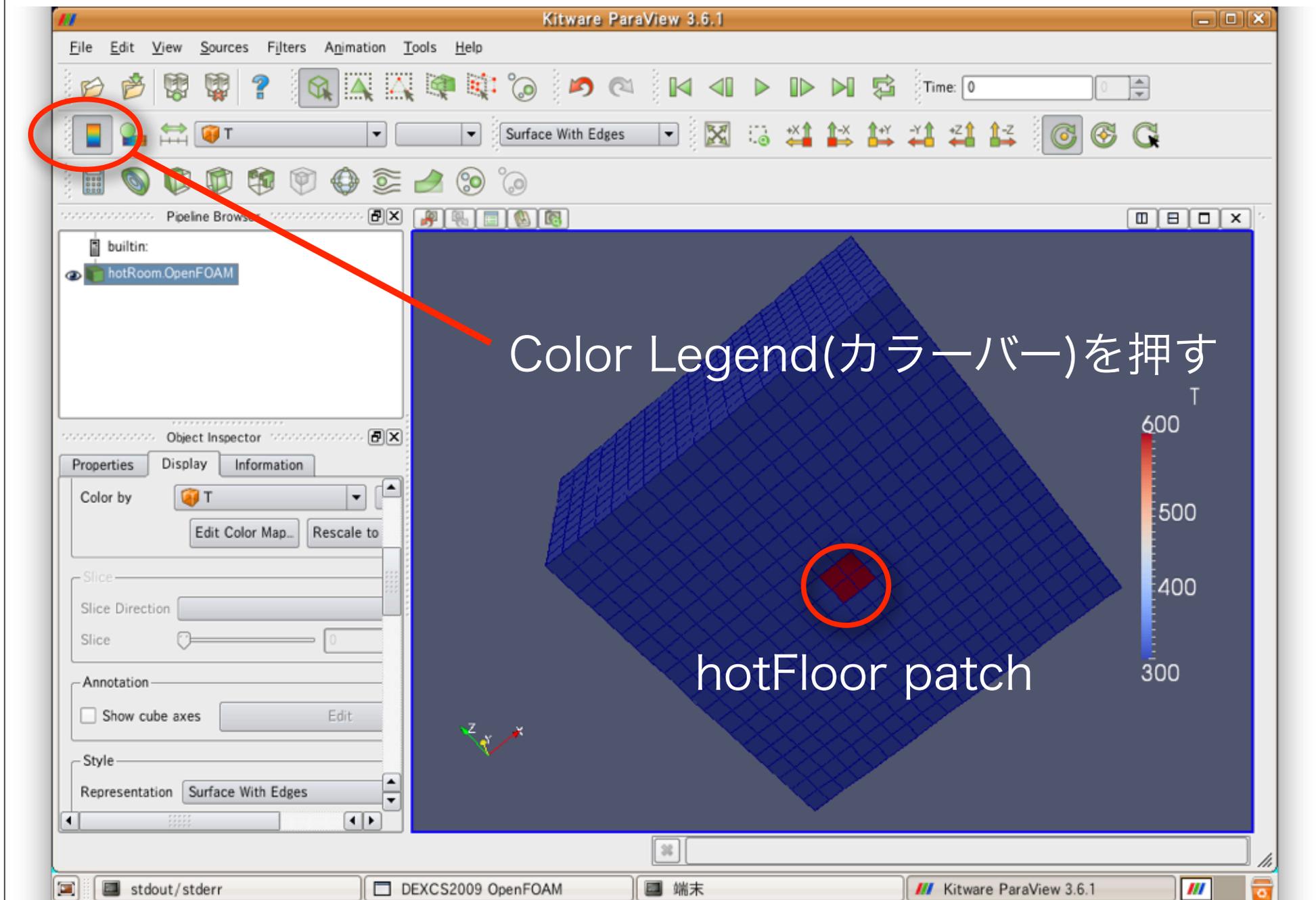




1. Displayタブを選ぶ

3. T(cell)を選ぶ

2. Representationにて
Surface With Edgesを選ぶ



6. 計算の実行

buoyantBoussinesqSimpleFoam ↵

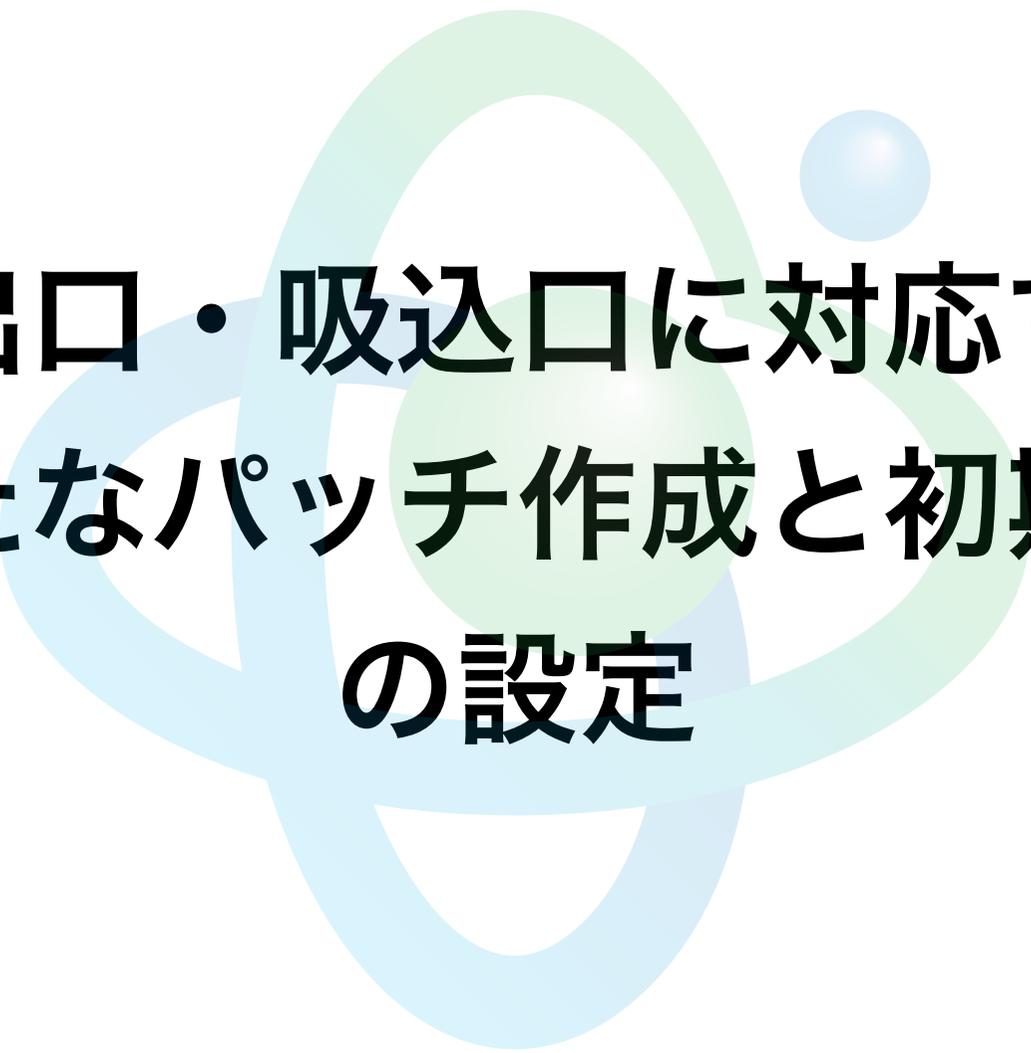
または

↓ logを残す場合

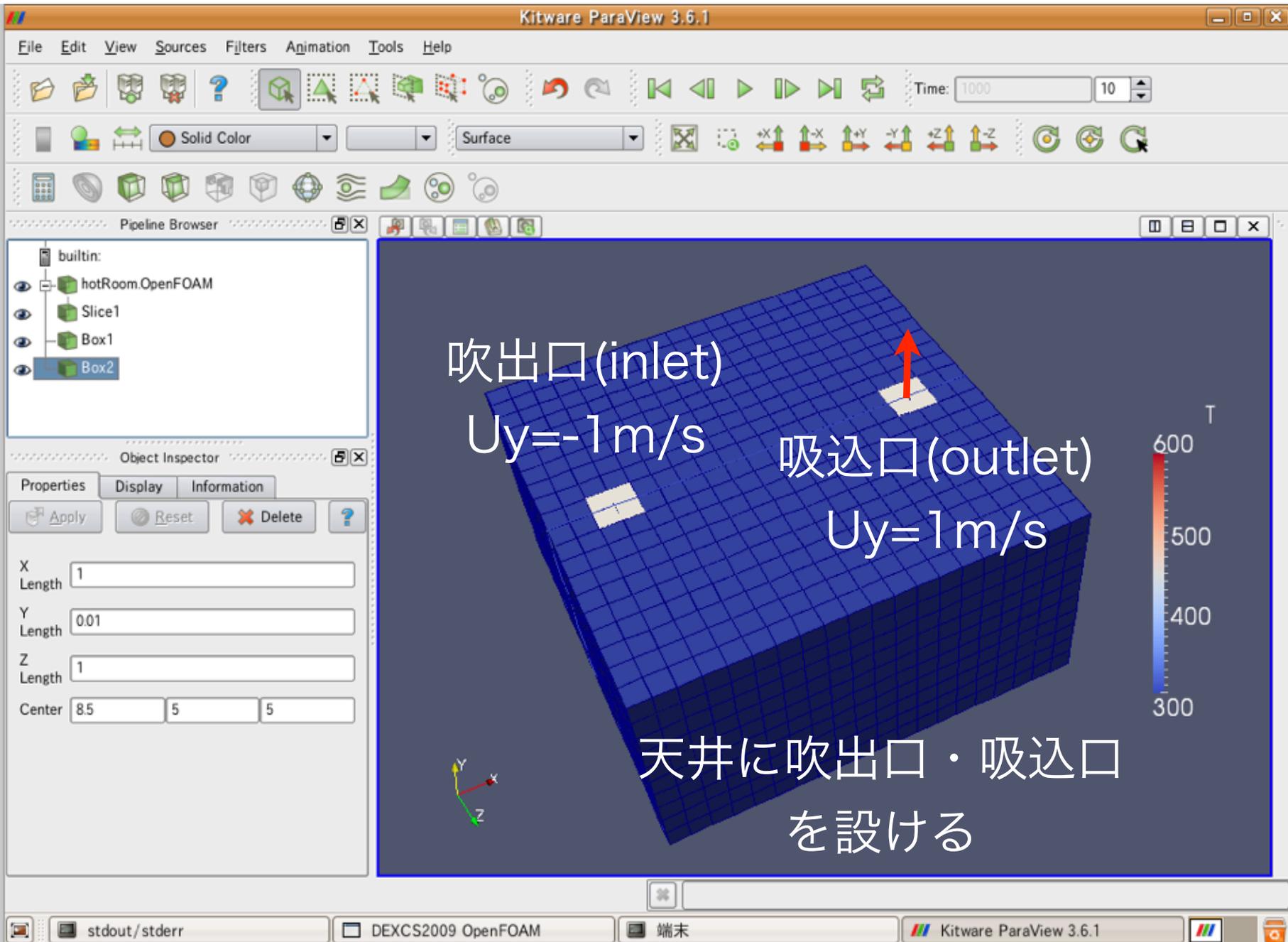
foamJob buoyantBoussinesqSimpleFoam ↵

tail -f log

↓ tailコマンドでlogを監視



**吹出口・吸込口に対応する
新たなパッチ作成と初期値
の設定**



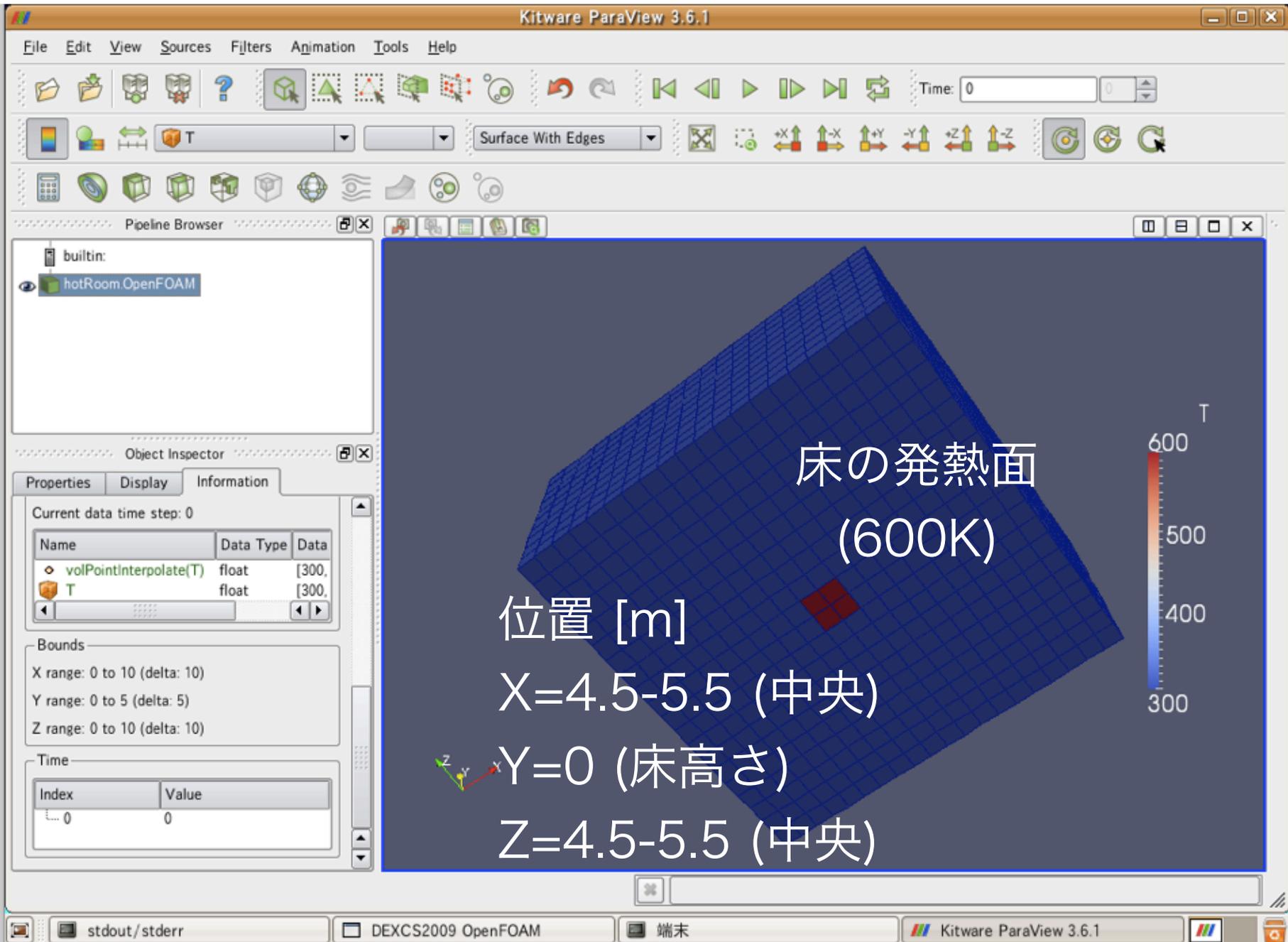
計算結果の消去

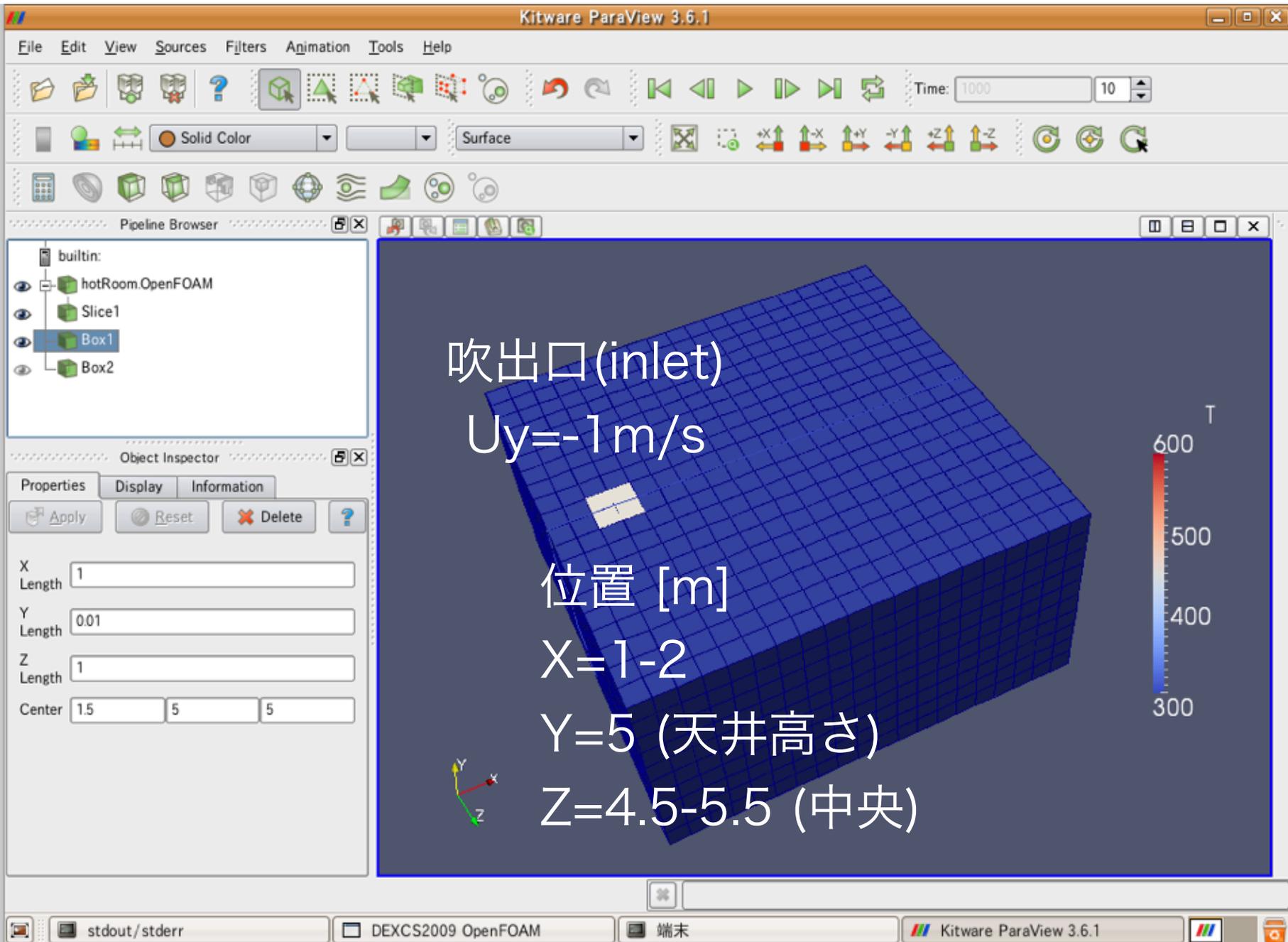
```
rm -r [1-9]* ↵
```

1. rmのrオプションは、ディレクトリごと消去
2. [1-9]は、1,2,...,9と展開される。[a-c]なら、a,b,c
3. [1-9]* は、0以外の計算結果のディレクトリ全てを表わす

faceSetDictの修正

```
gedit system/faceSetDict & ↵
```





faceSetDictの修正

```
object faceSetDict;
}
// ***** //
// Name of set to operate on
name inlet;
// One of clear/new/invert/add/delete/subset/list
action new;
// Actions to apply to pointSet. These are all the topoSetSource's ending
// in ..ToFace (see the meshTools library).
topoSetSources
(
  // Faces with face centre within box
  boxToFace
  {
    box (1 4.999 4.5) (2 5.001 5.5);
  }
);
// ***** //
```

faceSetの実行

faceSet ↩

出力

```
Reading faceSetDict
```

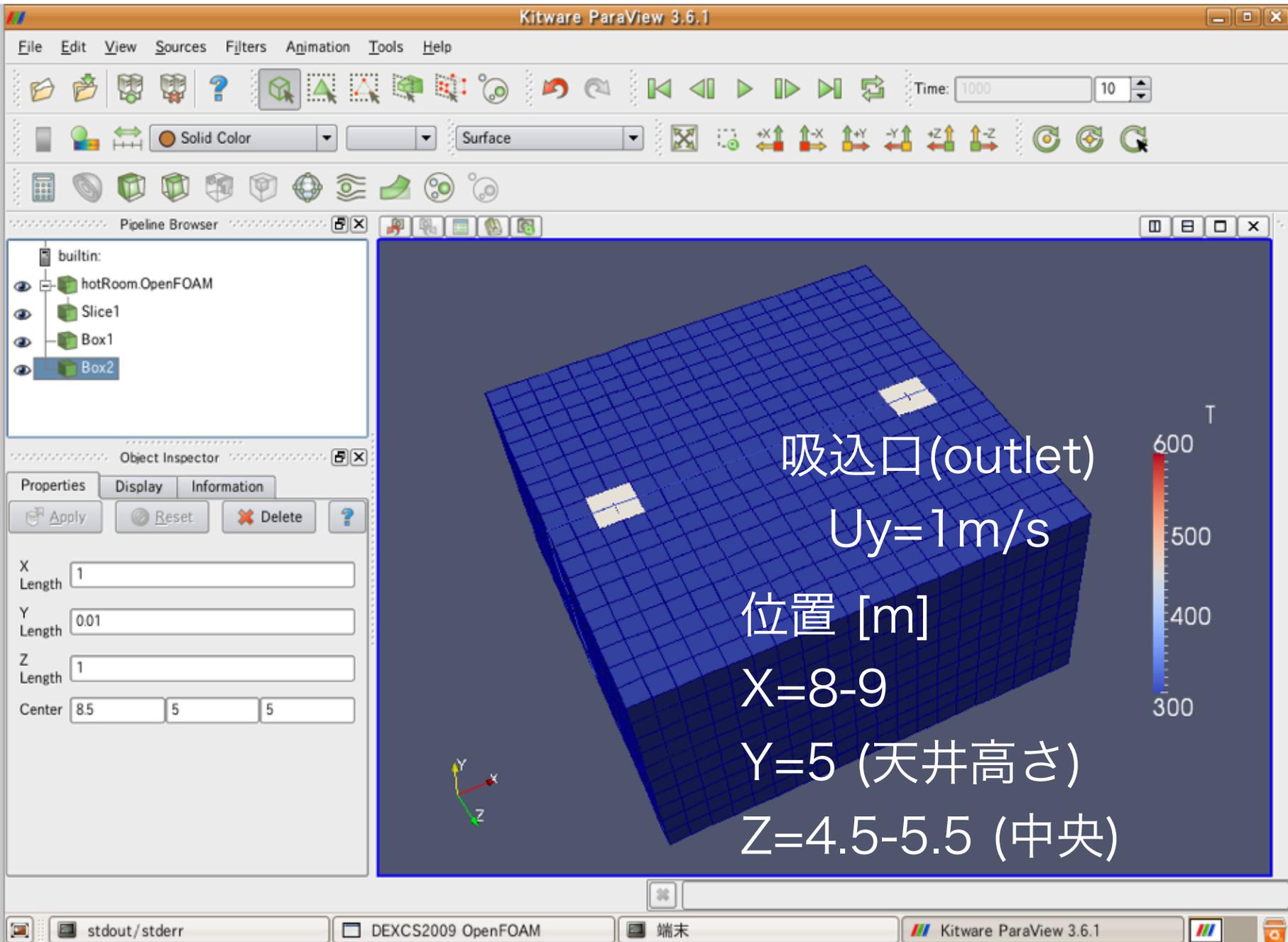
```
Backing up inlet into inlet_old
```

```
Set:inlet Size:0 Action:new
```

```
    Adding faces with centre within box (1 4.999 4.5) (2  
5.001 5.5)
```

```
Writing inlet (size 4) to "constant/polyMesh/sets/inlet"
```

4つの格子界面のfaceSetが、constant/polyMesh/sets/inletとして書き出された。



faceSetDictの修正

```
faceSetDict (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/heat.../buoyantBoussinesqSimpleFoam/hotRoom/system) - gedit
ファイル(F) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)
新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換
faceSetDict
object | faceSetDict;
}
// *****
// Name of set to operate on
name outlet;
// One of clear/new/invert/add/delete|subset/list
action new;
// Actions to apply to pointSet. These are all the topoSetSource's ending
// in ..ToFace (see the meshTools library).
topoSetSources
(
  // Faces with face centre within box
  boxToFace
  {
    box (8 4.999 4.5) (9 5.001 5.5);
  }
);
// *****
```

3. 「保存」を押す (Ctrl+Sでも可)

1.nameの値をoutletに変更

2. boxの領域を吸入口を囲む領域に変更する

(13行、14列) [挿入]

faceSetの実行

faceSet ↩

出力

```
Reading faceSetDict
```

```
Backing up outlet into outlet_old
```

```
Set:outlet Size:0 Action:new
```

```
    Adding faces with centre within box (8 4.999 4.5) (9  
5.001 5.5)
```

```
Writing outlet (size 4) to "constant/polyMesh/sets/outlet"
```

4つの格子界面のfaceSetが、constant/polyMesh/sets/outletとして書き出された。

createPatchDictの修正

createPatchDict (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/he...buoyantBoussinesqSimpleFoam/hotRoom/system) - gedit

ファイル(F) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)

新規 開く

1. 開く (Ctrl+Oでも可)

2. createPatchDictを選択

場所(P) 名前 最終変更日

場所(P)	名前	最終変更日
検索	controlDict	昨日の 21:24
最近開いたファイル	createPatchDict	昨日の 21:54
dexcs		
デスクトップ		
ファイル・システム		
Custom Live CD	fvSchemes	2010年06月23日
フロッピー・ドライブ	fvSolution	2010年06月23日

+ 追加(A) - 削除(B) 全てのファイル

エンコーディング(H): 自動検出

キャンセル(C) 開く(O)

// ***** //

(66行, 20列) [挿入]

createPatchDictの修正

```
set hotWall;
}
{
  name inlet;
  // Type of new patch
  dictionary
  {
    type patch;
  }
  constructFrom set;
  patches ();
  set inlet;
}
{
  name outlet;

  // Type of new patch
  dictionary
```

1. patch名(name)の値がinet
のブロックを追加

2. typeをpatchに変更

3. setの値を先程作成した
faceSet名のinletに変更

(74 行、23 列) [挿入]

createPatchDictの修正

```
patches ();
set inlet;
}
{
name outlet;
// Type of new patch
dictionary
{
type patch;
}
constructFrom set;
patches ();
set outlet;
}
);
// ***** //
```

3. 「保存」を押す(Ctrl+Sでも可)

1. patch名(name)の値がoutletのブロックを追加

2. typeをpatchに変更

3. setの値を先程作成したfaceSet名のoutletに変更

(88行、10列) [挿入]

createPatchの実行

```
createPatch -overwrite ↵
```

inletやoutletのfaceSetからなる新しいpatchである
inletやoutletがconstant/polyMeshに上書きで出力

各物理量の初期ファイルの修正

```
gedit 0/* & ↵
```

初期ファイル(0/T)の修正

```
T (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/heatTransfer/buoyantBoussinesqSimpleFoam/hotRoom/0) - gedit
ファイル(F) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)
新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換
T T.org T~ U U~ alphas alphas~ epsilon epsilon~ k k~ nut
object T;
1. Tのタグを選択
***** //
dimensions [0 0 0 1 0 0 0];
internalField uniform 300;
boundaryField
{
inlet
{
type fixedValue;
value uniform 290;
}
outlet
{
type zeroGradient;
}
floor
{
type fixedValue;
value uniform 300;
}
}
(24 行、10 列) [挿入]
```

初期ファイル(0/U)の修正

U (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/heatTransfer/buoyantBoussinesqSimpleFoam/hotRoom/U) - gedit

ファイル(E) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)

新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換

T T.org T U U alphas alphas epsilon epsilon k k nut

```
object      U;
}
// ***** //
dimensions  [0 1 -1 0 0 0];
internalField uniform (0 0 0);
boundaryField
{
    inlet
    {
        type      fixedValue;
        value      uniform (0 -1 0);
    }
    outlet
    {
        type      fixedValue;
        value      uniform (0 1 0);
    }
    floor
    {
        type      fixedValue;
```

1. Uのタグを選択

2. inletは (0 -1 0)

3. outletは (0 1 0)

(31 行、37 列) [挿入]

初期ファイル(0/kappat)の修正

object alphas;

```
}  
// *****  
dimensions [1 -1 -1 0 0 0];  
internalField uniform 0;  
boundaryField  
{  
  inlet  
  {  
    type zeroGradient;  
  }  
  outlet  
  {  
    type zeroGradient;  
  }  
  floor  
  {
```

1. kappatのタグを選択

2. inletとoutletのtypeはzeroGradient

(28行、8列) [挿入]

初期ファイル(0/k)の修正

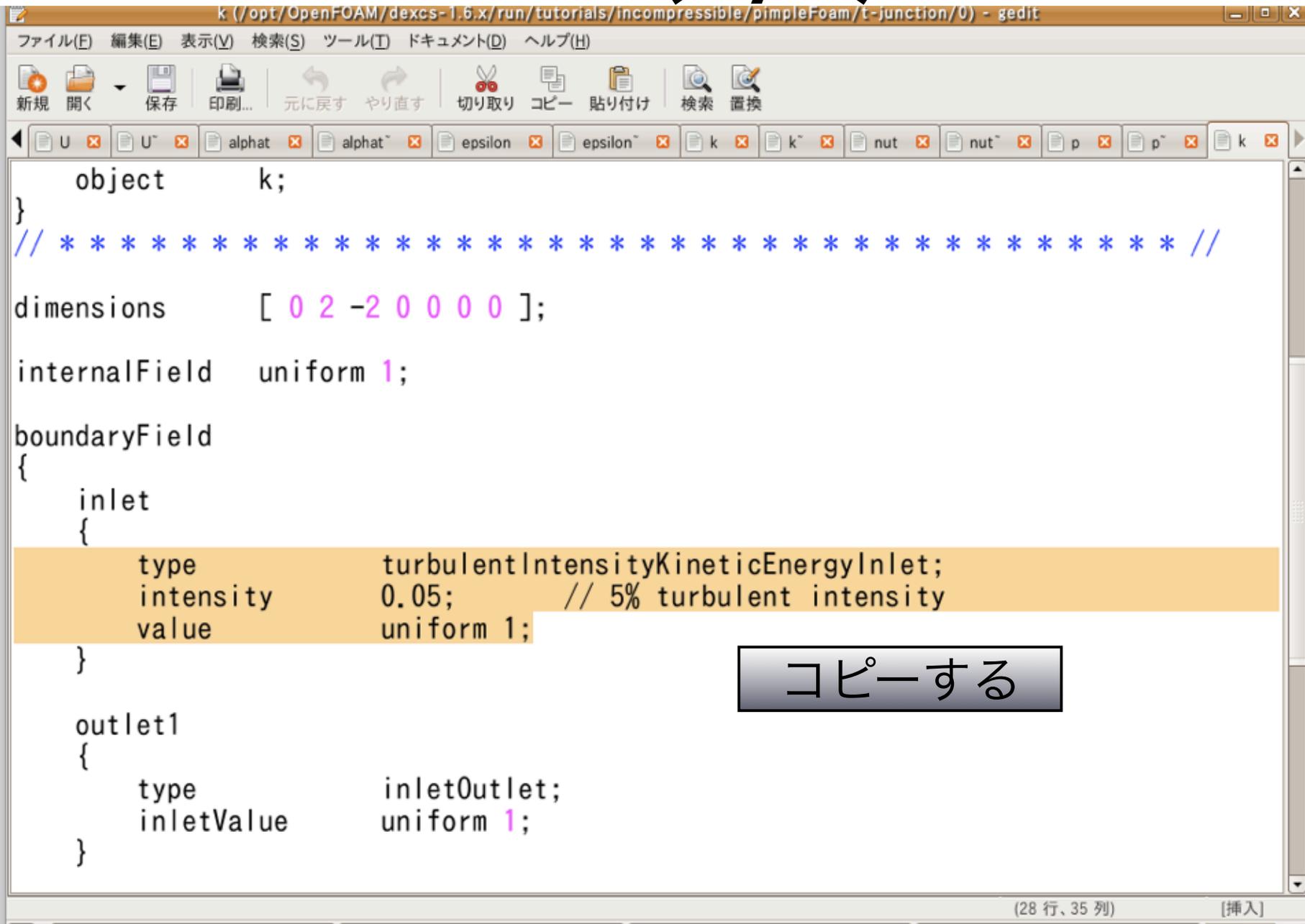
The image shows a file explorer window titled "ファイルを開く" (Open File) with the following path: `opt / OpenFOAM / dexcs-1.6.x / run / tutorials / incompressible / pimpleFoam / t-junction / 0`. The file `k` is selected in the list. A text box overlay contains the following text:

サンプルとして
tutorials/incompressible/pimpleFoam/
t-junction/0/k を開く

The background shows a code editor with the following text:

```
objec
}
// * * *
dimension
internalF
boundaryF
{
inlet
{
t
v
}
outle
{
t
}
t
v
```

初期ファイル(0/k)の修正



```
k (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/incompressible/pimpleFoam/t-junction/0) - gedit
ファイル(E) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)
新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換
U U~ alphas alphas~ epsilon epsilon~ k k~ nut nut~ p p~ k
object      k;
}
// *****
dimensions  [ 0 2 -2 0 0 0 ];
internalField  uniform 1;
boundaryField
{
  inlet
  {
    type          turbulentIntensityKineticEnergyInlet;
    intensity     0.05;          // 5% turbulent intensity
    value         uniform 1;
  }
  outlet1
  {
    type          inletOutlet;
    inletValue    uniform 1;
  }
}
```

コピーする

(28 行、35 列) [挿入]

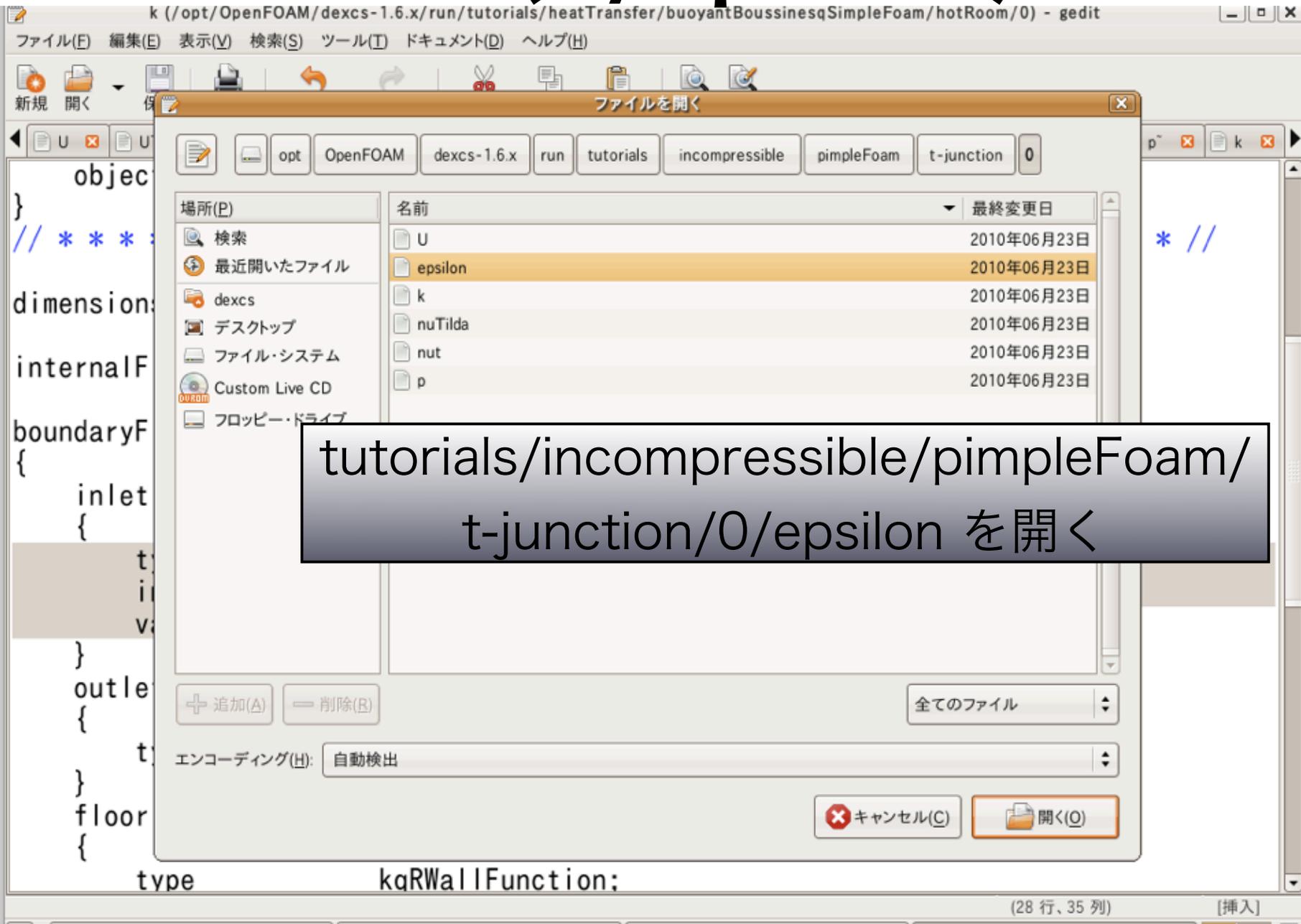
初期ファイル(0/k)の修正

The image shows a gedit editor window with a file named 'k' open. The file content is a turbulence model configuration. Annotations are provided in grey boxes:

- 1. kのタグを選択 (Select the k tag) - A red circle highlights the 'k' tag in the file's header.
- 2. intensityは0.05 (乱れ強さが5%) (intensity is 0.05 (turbulence intensity is 5%)) - A box points to the '0.05;' value in the 'intensity' field of the 'inlet' boundary condition.
- 3. outletはzeroGradient (outlet is zeroGradient) - A box points to the 'zeroGradient;' value in the 'outlet' boundary condition.

```
object      k;
}
// ***** //
dimensions  [0 2 -2 0 0 0];
internalField  uniform 0.1;
boundaryField
{
  inlet
  {
    type      turbulentIntensityKineticEnergyInlet;
    intensity  0.05; // 5% turbulent intensity
    value     uniform 1;
  }
  outlet
  {
    type      zeroGradient;
  }
  floor
  {
    type      kqRWallFunction;
  }
}
```

初期ファイル(0/epsilon)の修正



初期ファイル(0/epsilon)の修正

```
object      epsilon;
}
// *****
dimensions  [ 0 2 -3 0 0 0 ];
internalField  uniform 1;
boundaryField
{
  inlet
  {
    type          turbulentMixingLengthDissipationRateInlet;
    mixingLength  0.01;          // 1cm - half channel height
    value         uniform 1;
  }
  outlet1
  {
    type          inletOutlet;
    inletValue    uniform 1;
  }
}
```

コピーする

(28 行、35 列) [挿入]

初期ファイル(0/epsilon)の修正

```
object
{
  epsilon;
}
// *****

dimensions      [0 2 -3 0 0 0 0];
internalField   uniform 0.01;
boundaryField
{
  inlet
  {
    type          turbulentMixingLengthDissipationRateInlet;
    mixingLength  0.1; // 10cm
    value         uniform 1;
  }
  outlet
  {
    type          zeroGradient;
  }
  floor
  {
    type          epsilonWallFunction;
  }
}
```

1. epsilonのタグを選択

2. mixingLengthは0.1
(乱流渦長さスケールを
吹出口幅の1/10)

3. outletはzeroGradient

初期ファイル(0/nut)の修正

nut (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/heatTransfer/buoyantBoussinesqSimpleFoam/hotRoom/0) - gedit

ファイル(F) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)

新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換

alphat x alphat~ x epsilon x epsilon~ x k x k~ x nut x nut~ x p x p~ x k x epsilon x

```
object nut;
}
// ****
dimensions [0 2 -1 0 0 0];
internalField uniform 0;
boundaryField
{
  inlet
  {
    type zeroGradient;
  }
  outlet
  {
    type zeroGradient;
  }
  floor
  {
    type nutWallFunction;
    value uniform 0;
  }
}
```

1. nutのタグを選択

2. inletとoutletのtypeはzeroGradient

(30行、36列) [挿入]

初期ファイル(0/p_rgh)の修正

p (/opt/OpenFOAM/dexcs-1.6.x/run/tutorials/heatTransfer/buoyantBoussinesqSimpleFoam/hotRoom/0) - gedit

ファイル(E) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)

新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換

alphan alpha~ epsilon epsilon~ k k~ nut nut~ p k epsilon

```
object
}
// *****
dimensions      [0 2 -2 0 0 0];
internalField    uniform 0;
boundaryField
{
    inlet
    {
        type      zeroGradient;
    }
    outlet
    {
        type      zeroGradient;
    }
    floor
    {
        type      buoyantPressure;
        rho       rhok;
        value     uniform 0;
    }
}
```

1. p_rghのタグを選択

2. inletとoutletのtypeはzeroGradient

(20行、1列) [挿入]

その他の初期ファイルの修正

The screenshot shows a gedit window with the following menu and toolbar items:

- メニュー: ファイル(F) 編集(E) 表示(V) 検索(S) ツール(T) ドキュメント(D) ヘルプ(H)
- ツールバー: 新規 開く 保存 印刷... 元に戻す やり直す 切り取り コピー 貼り付け 検索 置換
- タブ: *T *U *alphan *epsilon *k *nut *p

The main text area contains the following code:

```
boundaryField
{
    floor
    {
        type
        rho
        value
    }
    hotFloor
    {
        type
        rho
        value
    }
    ceiling
    {
        type
        rho
        value
    }
}
```

The dialog box contains the following text:

まだ保存していないドキュメントが 7個あります。閉じる前にそれらを保存しますか？

保存するドキュメントを選択して下さい(E):

- T
- U
- alphan
- epsilon
- k
- nut

保存しないと変更した内容が完全に失われます。

Buttons: 保存せずに閉じる(W) キャンセル(C) **保存(S)**

Bottom text box: Ctrl+Qで終了し、変更を全て保存

Status bar: (24 行, 6 列) [挿入]

計算実行

```
buoyantBoussinesqSimpleFoam ↵
```

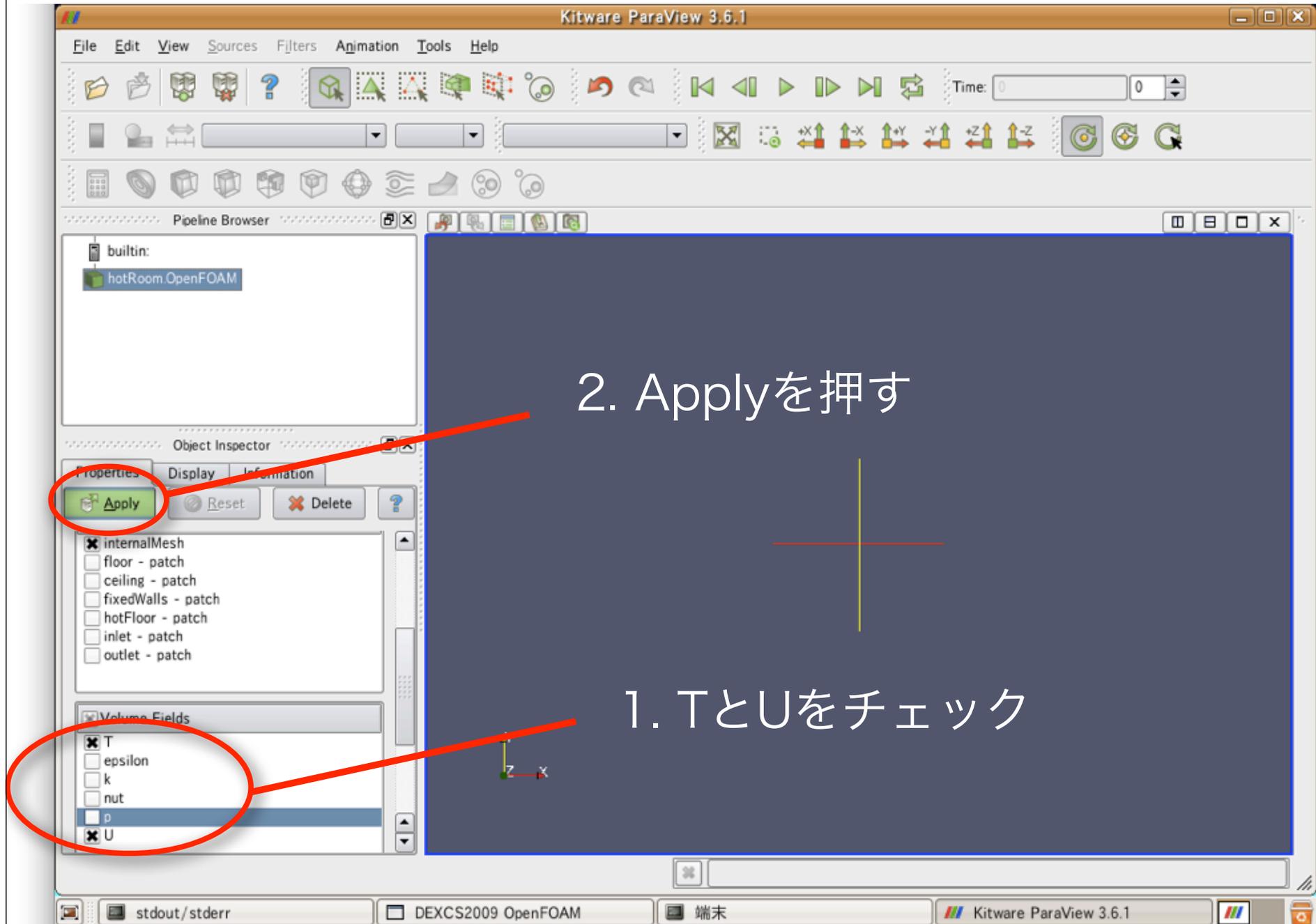
または

```
foamJob buoyantBoussinesqSimpleFoam ↵
```

```
tail -f log
```

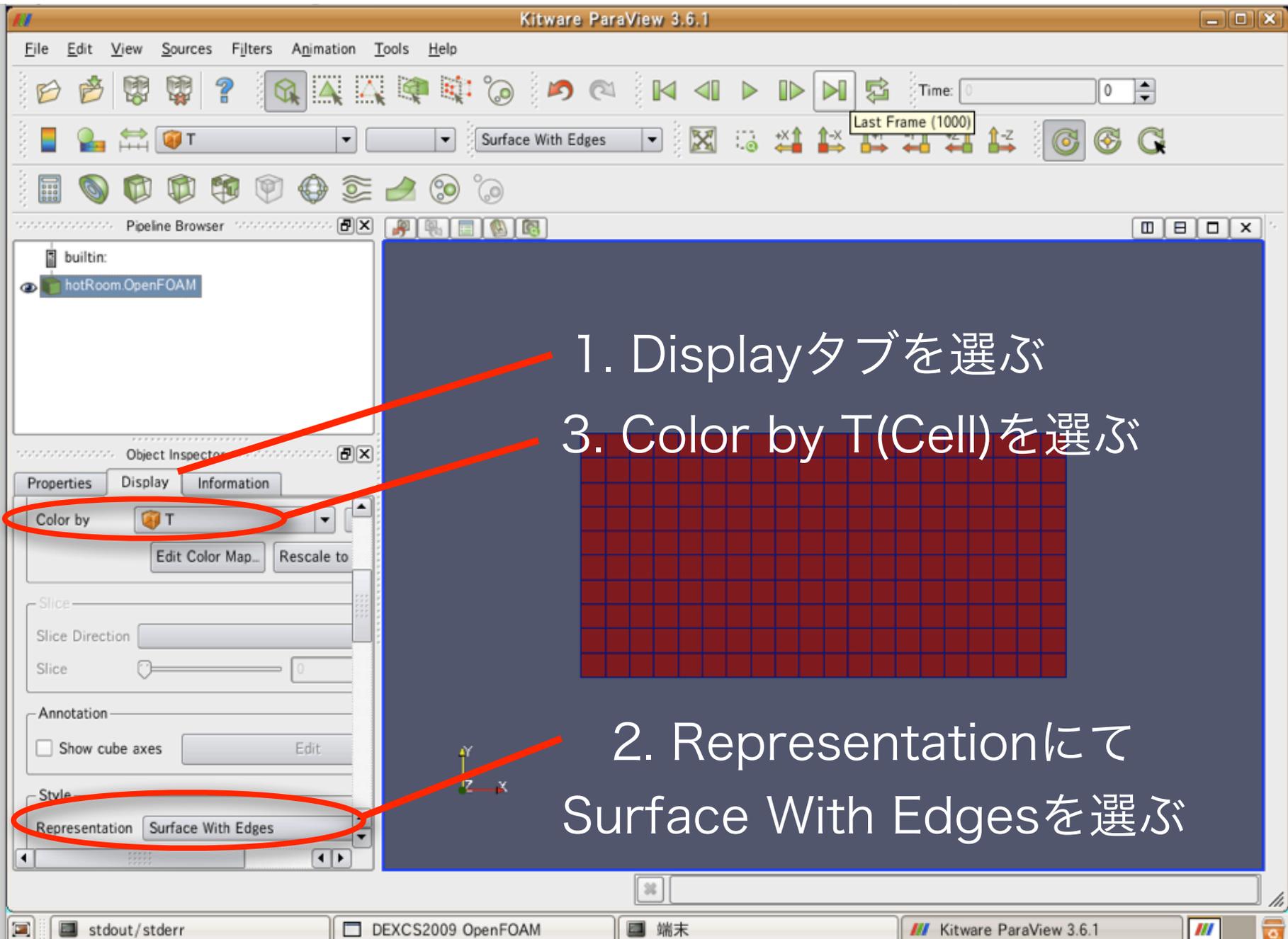
結果の可視化

```
paraFoam & ↵
```



2. Applyを押す

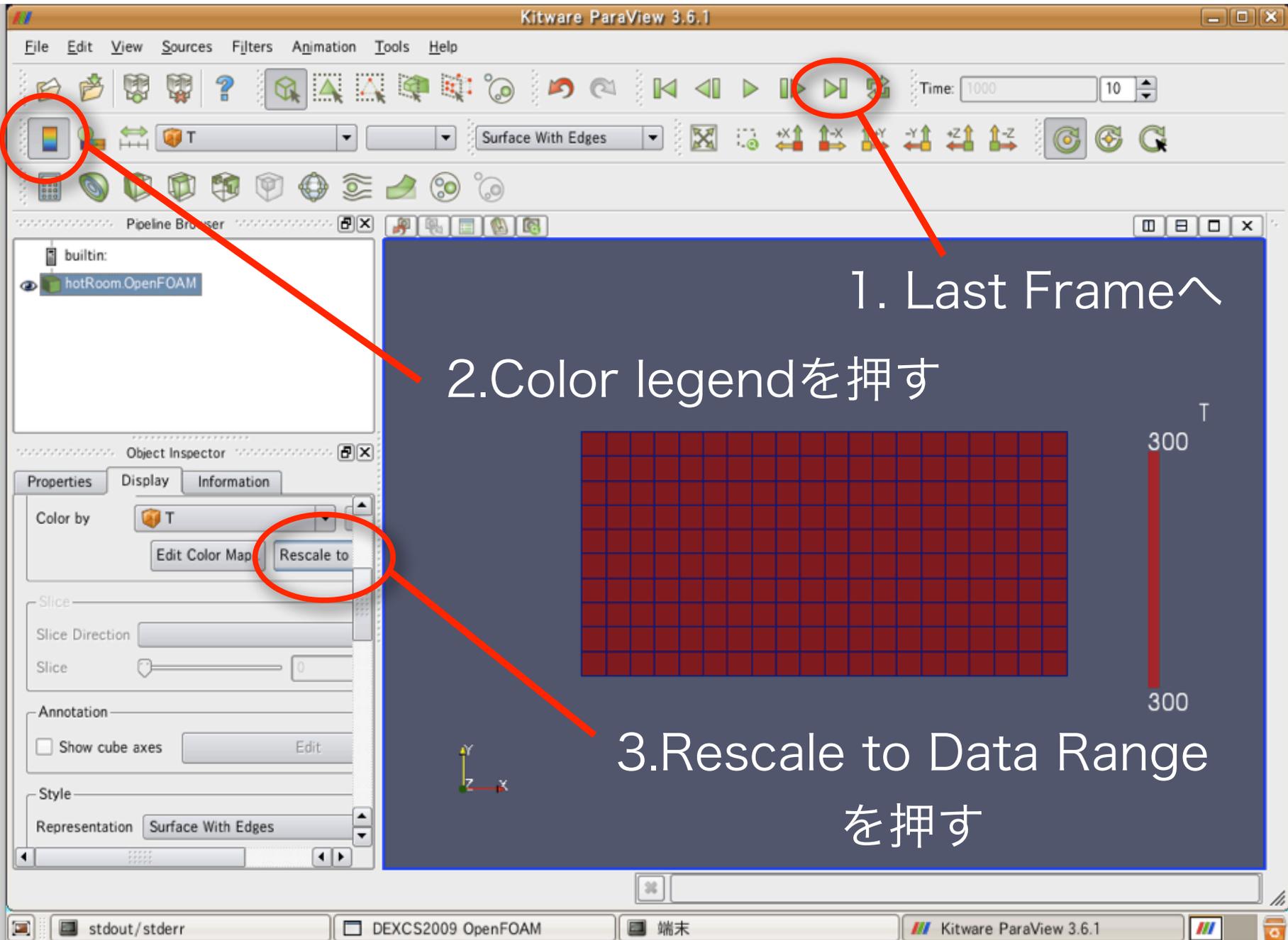
1. TとUをチェック



1. Displayタブを選ぶ

3. Color by T(Cell)を選ぶ

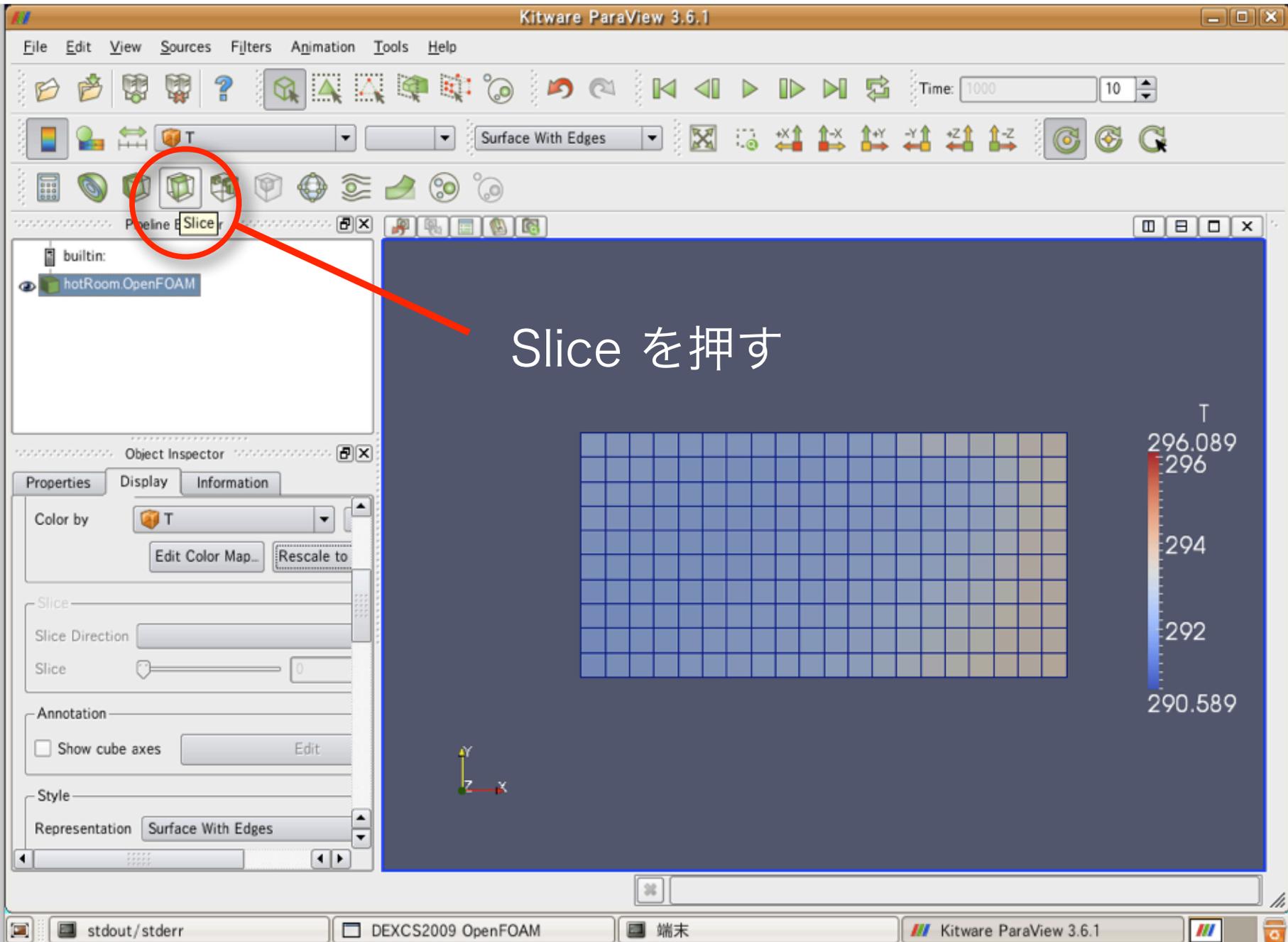
2. Representationにて
Surface With Edgesを選ぶ



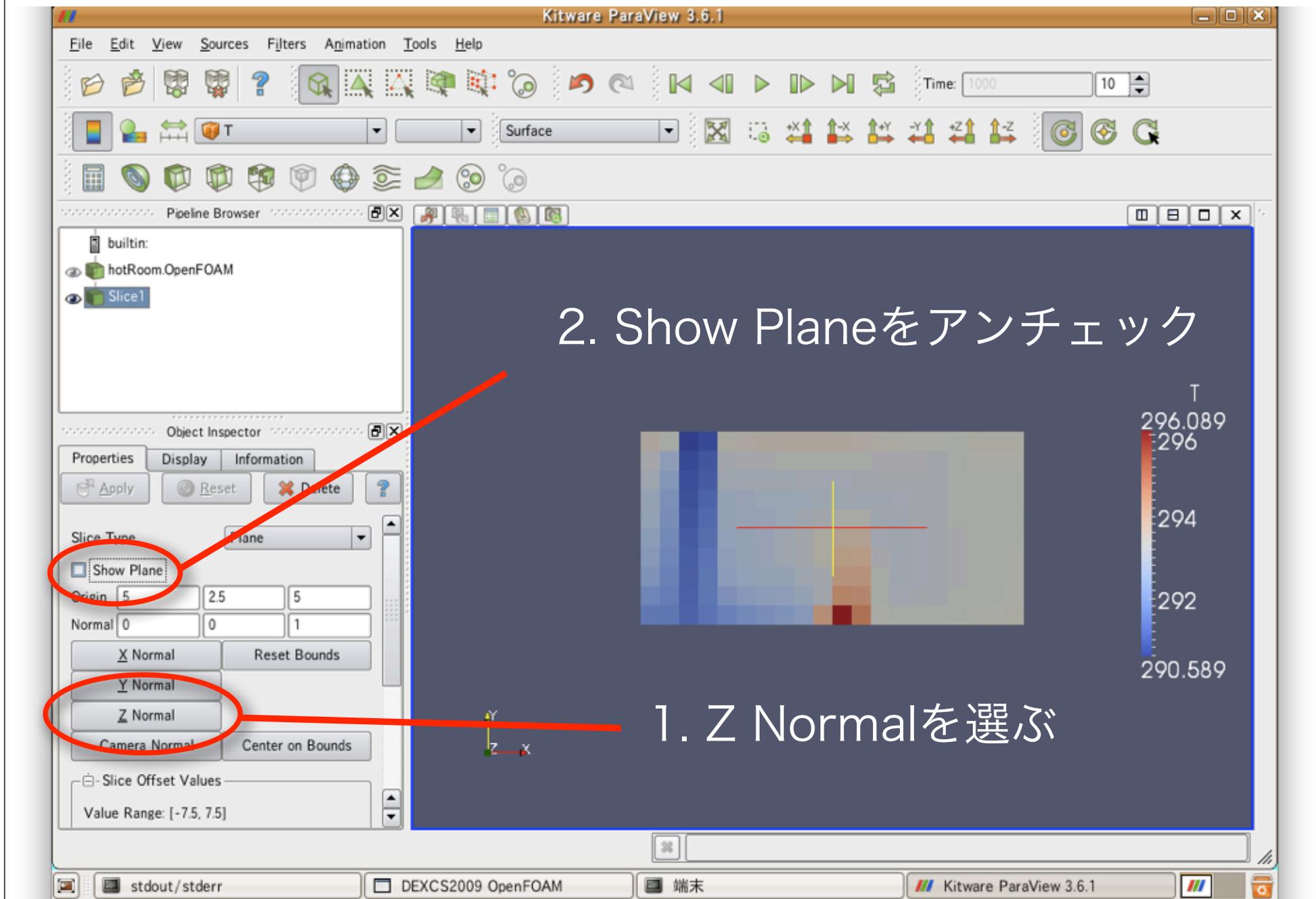
1. Last Frame^

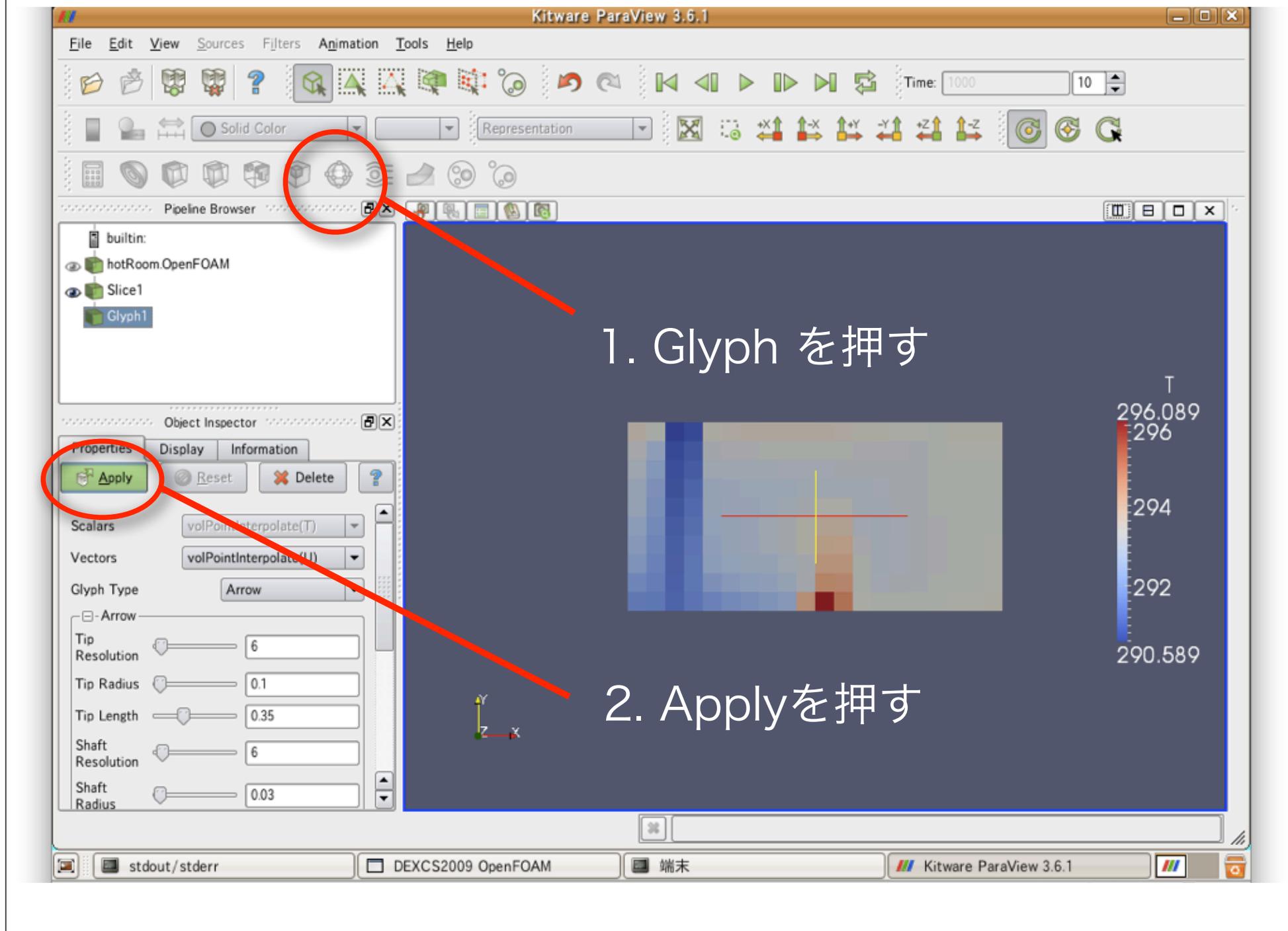
2. Color legendを押す

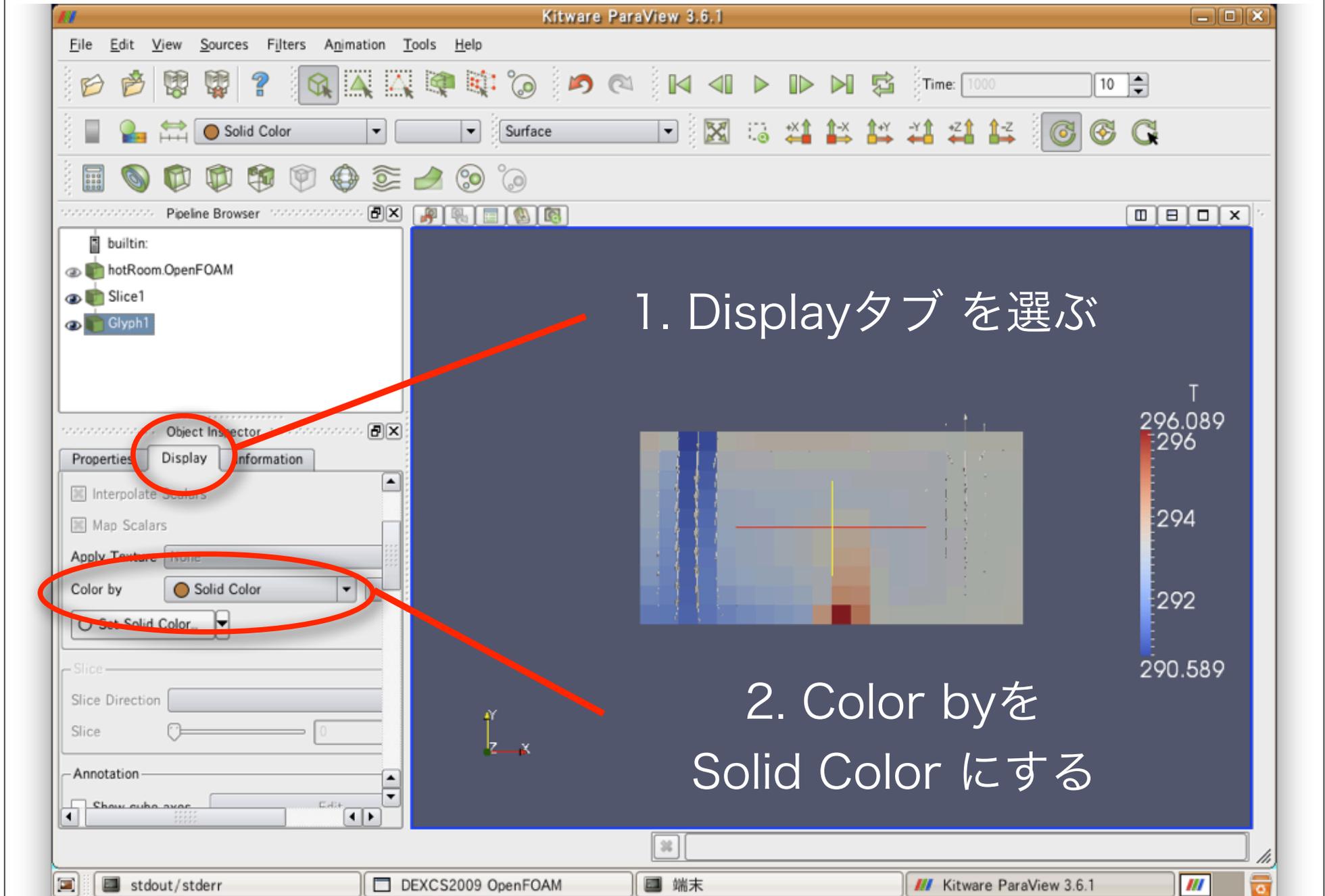
3. Rescale to Data Range
を押す



Slice を押す

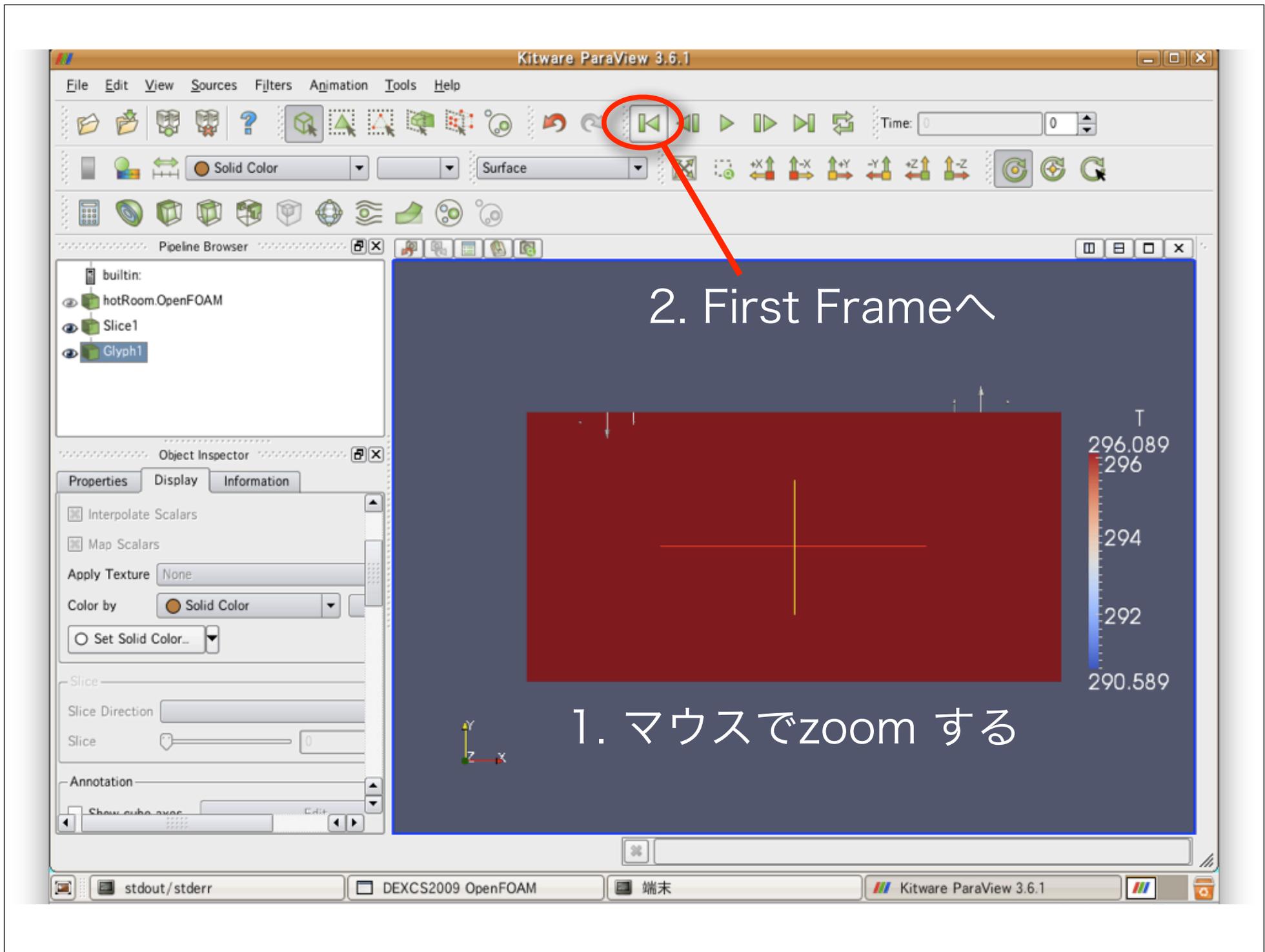


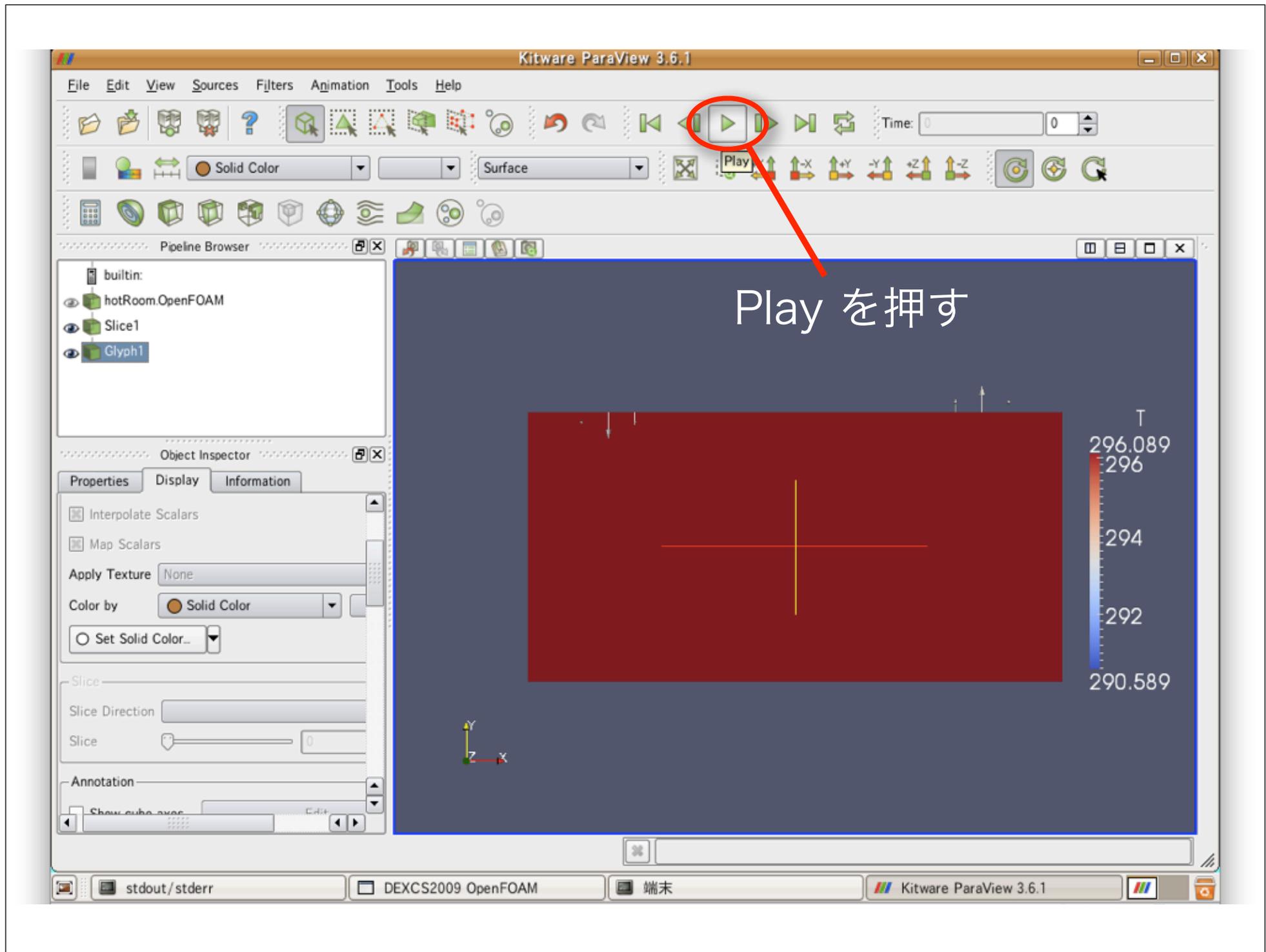


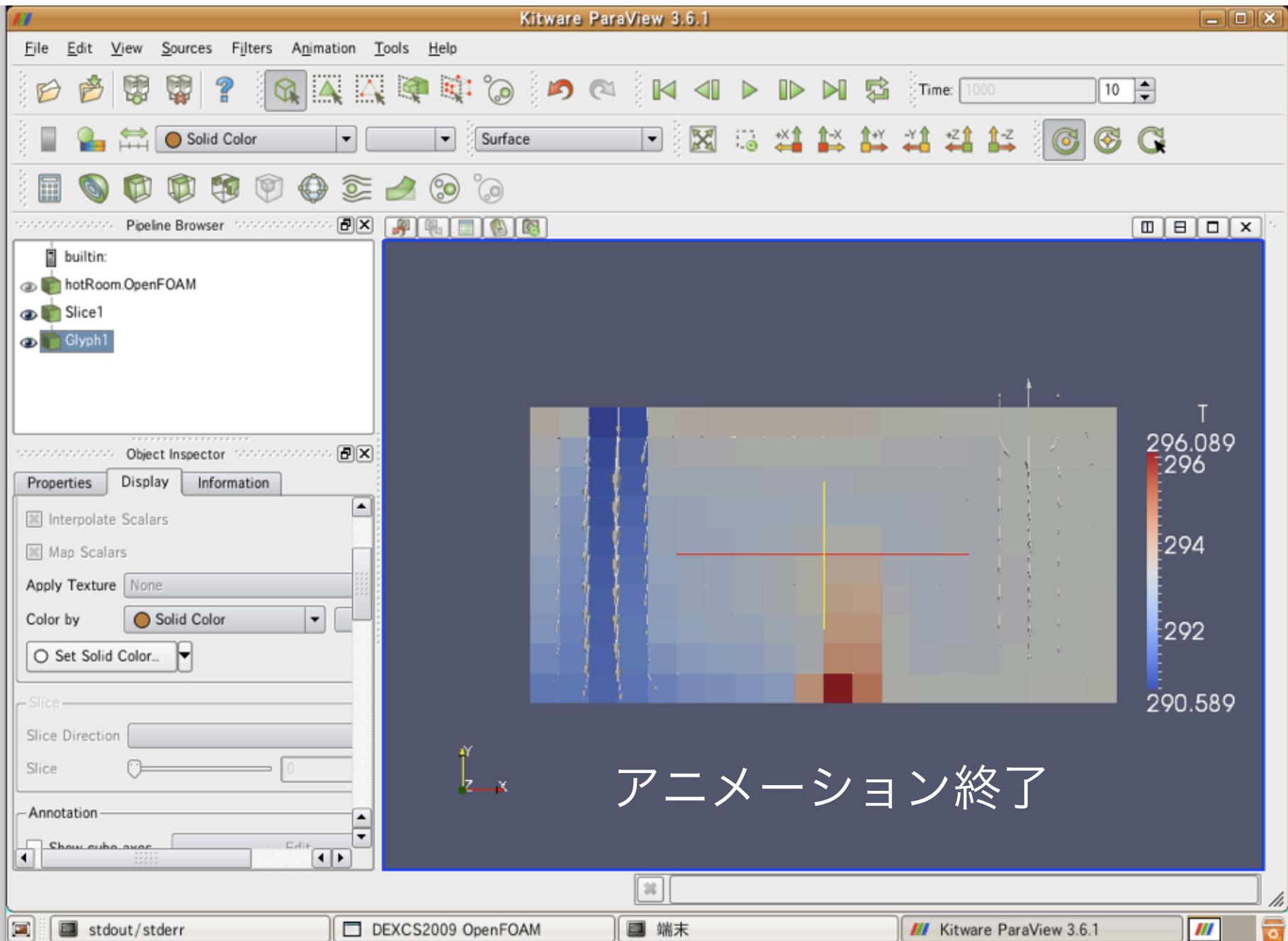


1. Displayタブを選ぶ

2. Color byを
Solid Colorにする







アニメーション終了

目次

1. 発熱面に対応する新たなパッチ作成と初期値の設定
2. 吹出口・吸込口に対応する新たなパッチ作成と初期値の設定
3. 質疑