OpenFOAM Study Group for beginner

monthly meeting 2012/03/24

Shinya Ogata
小縣信也
Twitter @oga_shin

Today's topic

- How to make setFields using cellSets
- Orignal utility "secondCellID"
- Coupled Heat and Moisture transport using customized chtMultiRegionFOAM

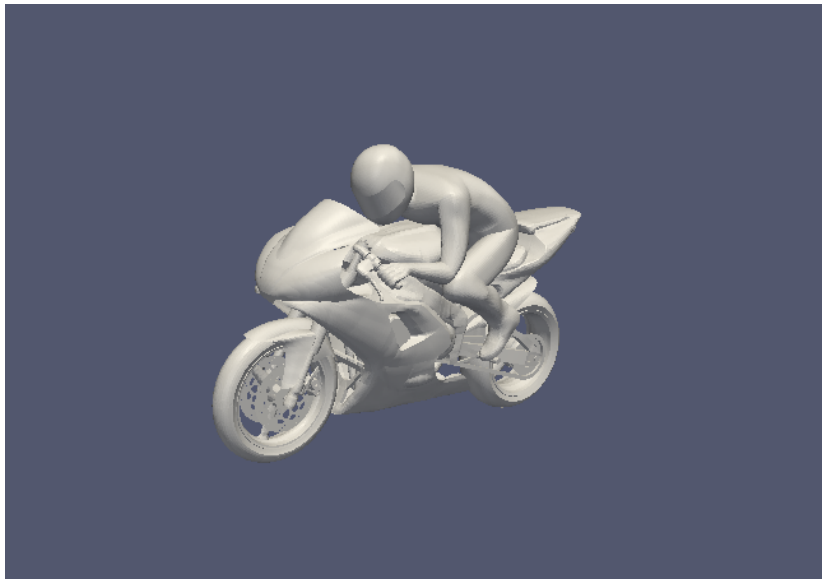
How to make setFields using cellSets

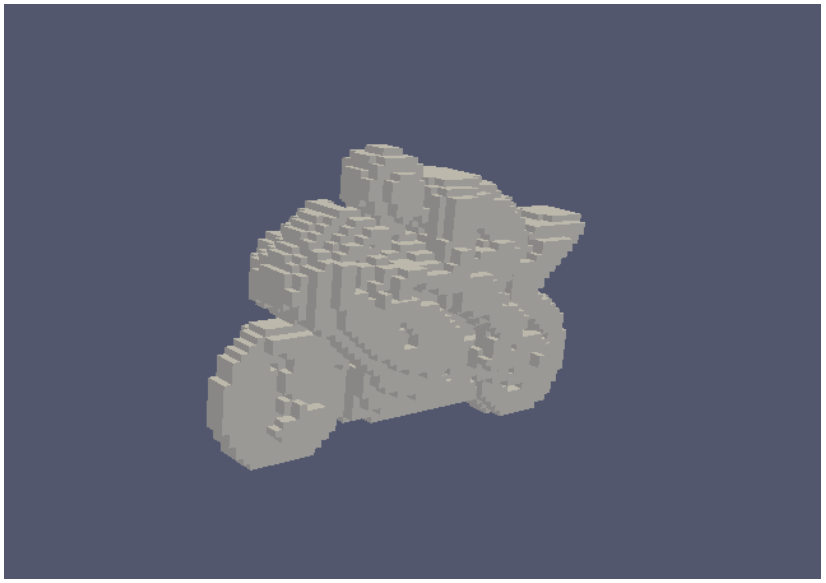
1. Make cellSets using STL file

Ex.

$setSet –batch bike.set

```
//bike.set
cellSet bike new surfaceToCell "constant/triSurface/
motorBike.stl" ((-1 -1 1.8)) true true false -1 -100
```





How to make cellSet usign STL file

/opt/openfoam171/applications/utilities/mesh/manipulation/cellSet/cellSetDict

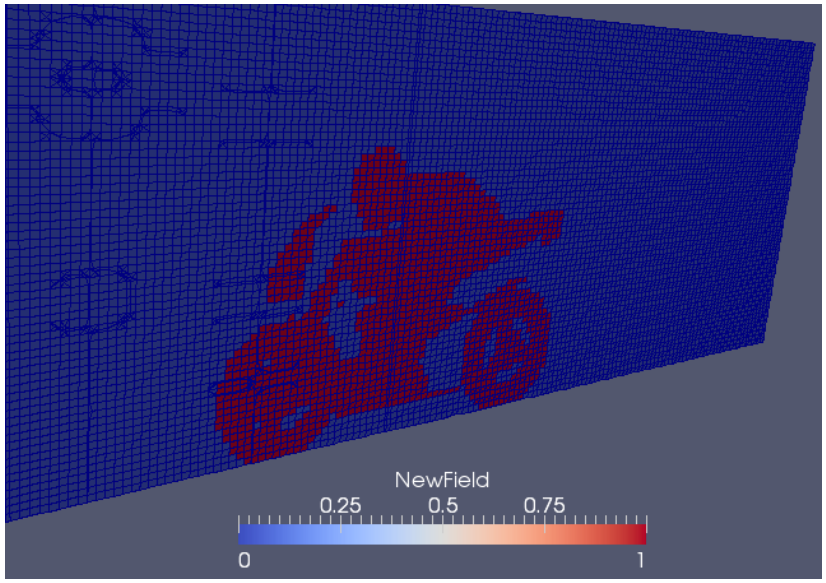
```
// Select based on surface
    surfaceToCell
    {
        file            "www.avl.com-geometry.stl";
        outsidePoints   ((-99 -99 -59));    // definition of outside
        includeCut      false;              // cells cut by surface
        includeInside   false;              // cells not on outside of surf
        includeOutside  false;              // cells on outside of surf
        nearDistance    -1;                 // cells with centre near surf
                                            // (set to -1 if not used)
        curvature       0.9;                // cells within nearDistance
                                            // and near surf curvature
                                            // (set to -100 if not used)
    }
```

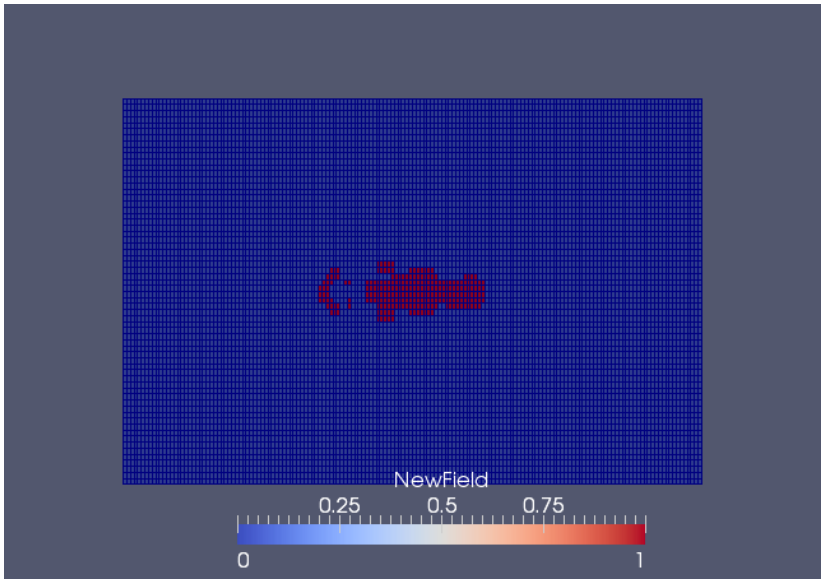
How to make setFields using cellSets

2. Run funkySetFileds

Ex.

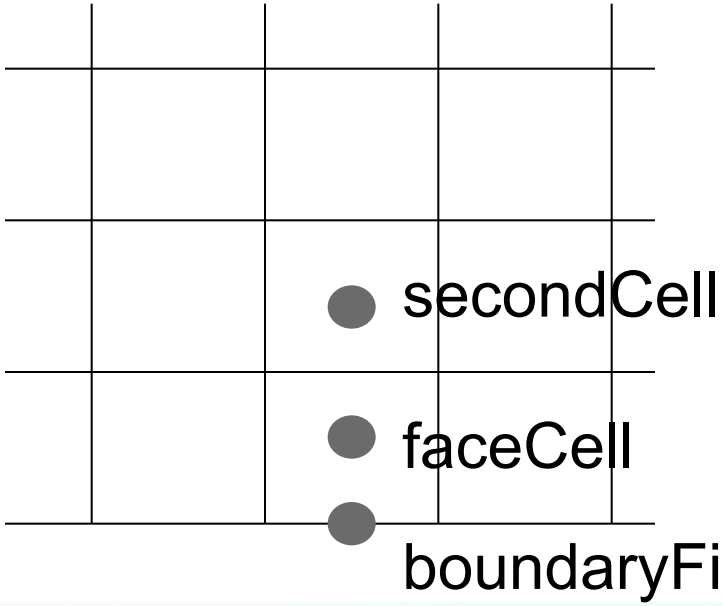
$funkySetFields -create -dimension "[1 -3 -1 0 0 0 0]" -time 0 -field NewField -expression "set(bike) ? 1 : 0 " >> log.funkySetFields



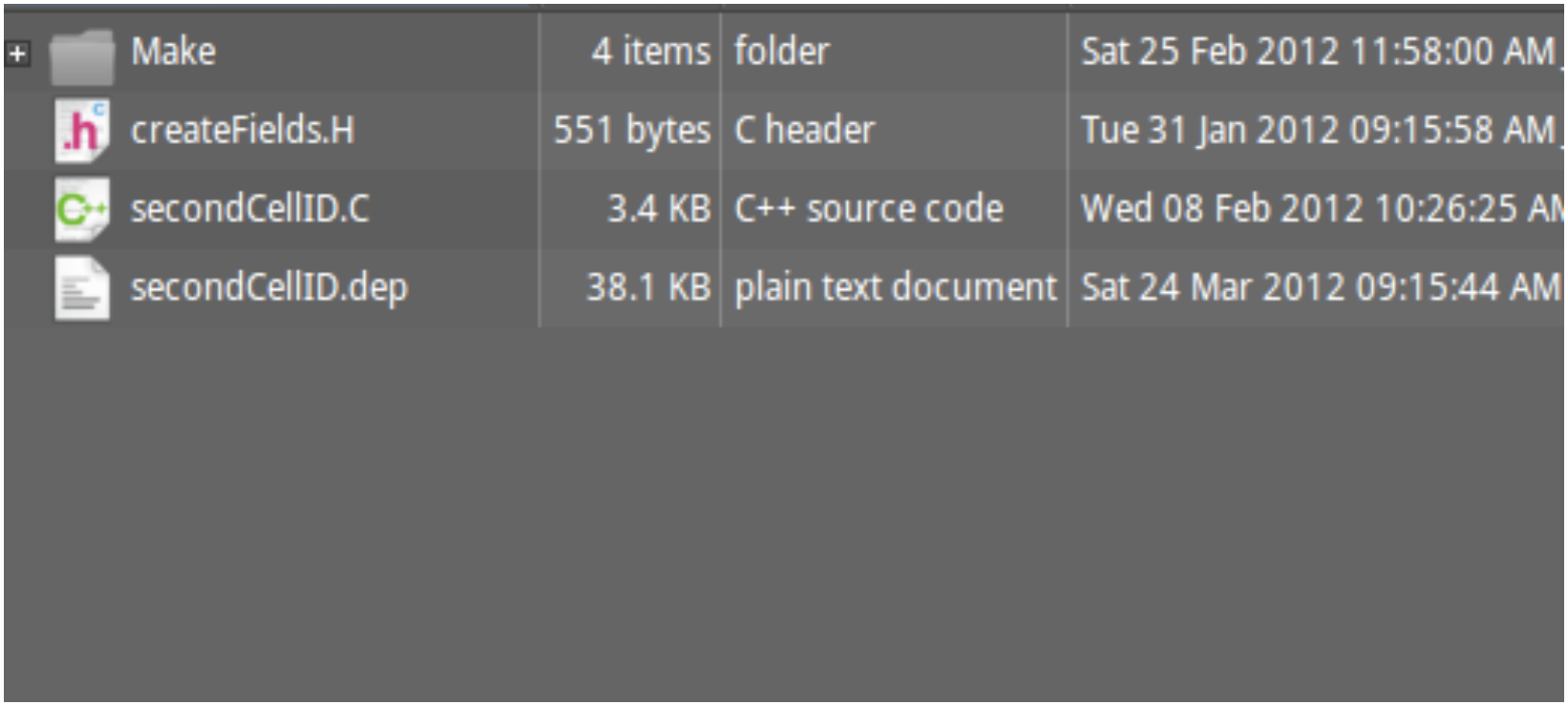


original utility “secondcelllD”

- I wanted to define boundaryFiled of next step using secondCell and boundaryField.
- However OpenFAOM has classes of faceCell and boundaryField, it doesn’t have a class about secondCell .
- I made a utility “secondCelllD” to get field value in secondCells.



Files of utility “secondcellID”



Files of utility “secondcellID”

createFields.H

```
Info<< "Reading field cellID\n" << endl;
    volScalarField cellID
    (
        IOobject
        (
            "cellID",
            runTime.timeName(),
            mesh,
            IOobject::MUST_READ,
            IOobject::AUTO_WRITE
        ),
        mesh
    );
    Info<< "Reading field sCellID\n" << endl;
    volScalarField scellID
    (
        IOobject
        (
            ”scellID",
            runTime.timeName(),
            mesh,
            IOobject::MUST_READ,
            IOobject::AUTO_WRITE
        ),
        mesh
    );
```

Files of utility “secondcellID”

secondcellID.C

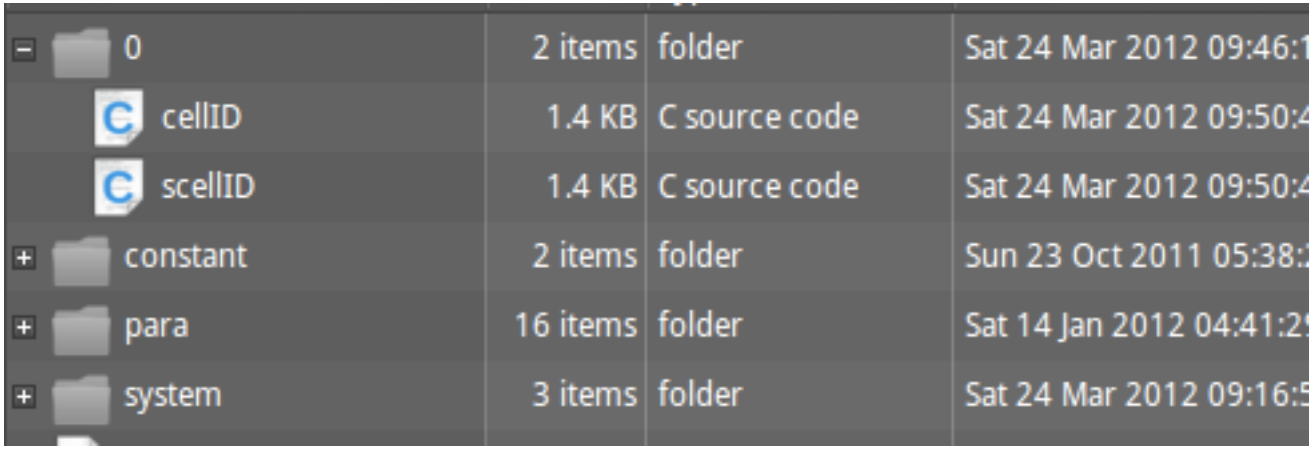
```
forAll(cellID, cellI){
        cellID[cellI]=cellI; //for confirmation
}

double Cfx ; //coordinate of patch center
double Cfy ; //coordinate of patch center
double Cfz ; //coordinate of patch center
double secondCellID ;

forAll(mesh.boundary(), q){
  if (q == 4 ){
        forAll(mesh.boundary()[q], j){
           Cfx = mesh.Cf().boundaryField()[q][j].x(); //coordinate of patch center
           Cfy = mesh.Cf().boundaryField()[q][j].y(); //coordinate of patch center
           Cfz = mesh.Cf().boundaryField()[q][j].z(); //coordinate of patch center
           Cfz += 0.375;  //reference distance
           secondCellID = mesh.findCell(point(Cfx, Cfy, Cfz));
           scellID[mesh.boundary()[q].faceCells()[j]] = secondCellID;
        }
    }
  }
```

How to use the utility “secondcellID”

- *Make scellID and cellID under 0 directory*
  scellID and cellID are setted properly.

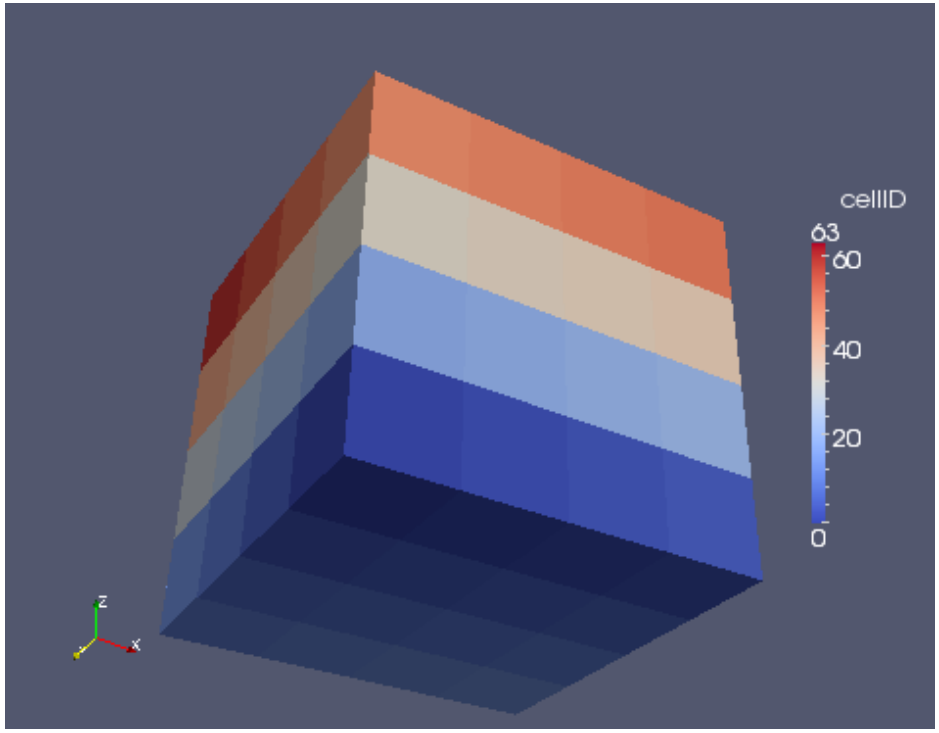


- Run the utility
  $secondCellID

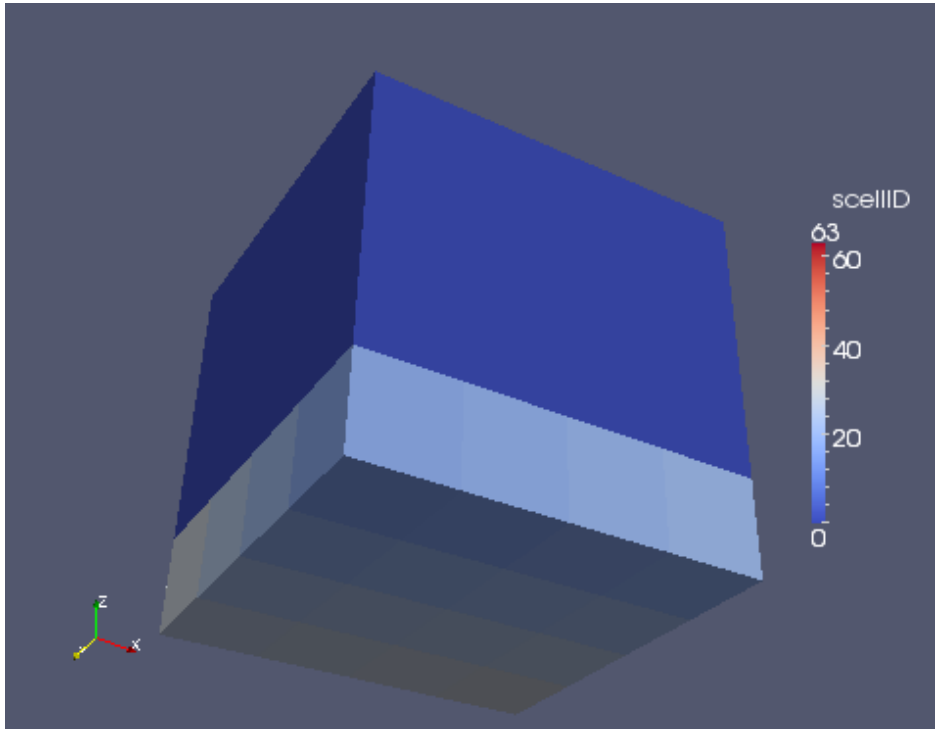
- Check result
  $paraFoam

original utility “secondcellID”

cellID



scellID



How to use secondcellID in a solver

```
// in a solver
volScalarField scellID
    (
        IOobject
        (
            ”scellID",
            runTime.timeName(),
            mesh,
            IOobject::MUST_READ,
            IOobject::AUTO_WRITE
        ),
        mesh
    );

forAll(mesh.boundary(), q){
        forAll(mesh.boundary()[q], j){
                cellID = scellID[mesh.boundary()[q].faceCells()[j]];
                deltaT = T.boundaryField()[q][j] - T[cellID];
                •
                •
                •
        }
}
```