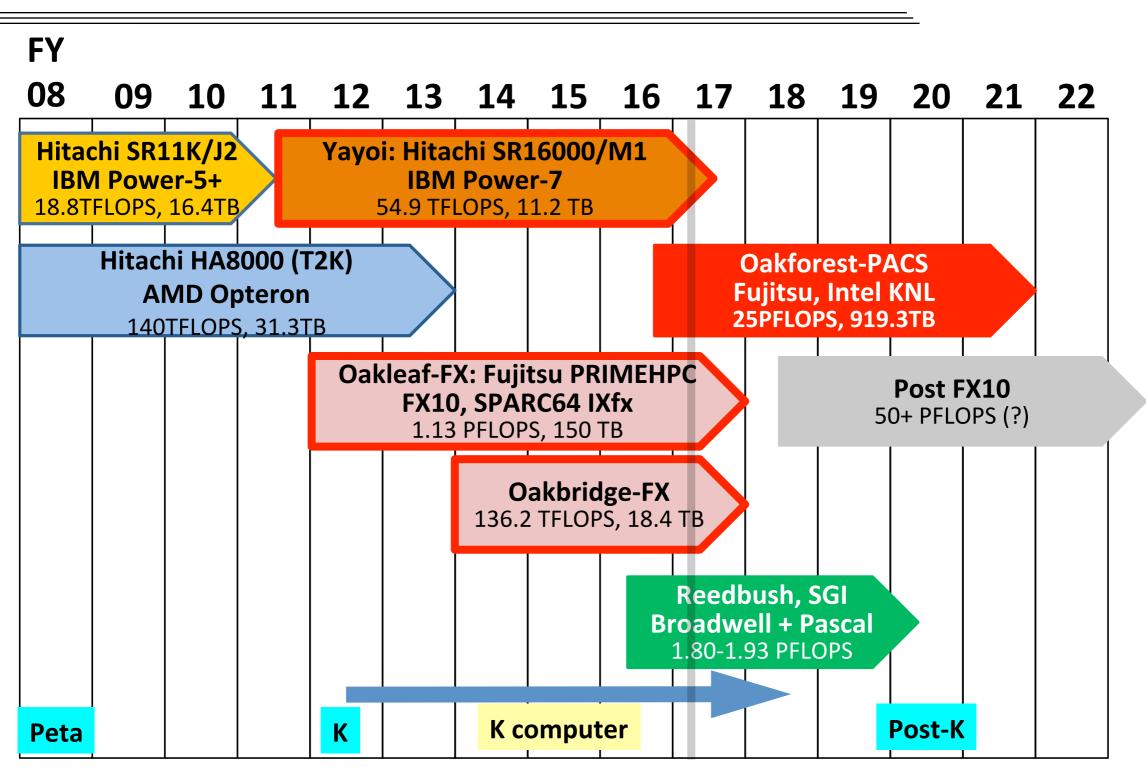
### 東京大学情報基盤センター お試しアカウント付き並列プログラミング講習会「OpenFOAM入門」 2017年5月12日@東京大学情報基盤センター遠隔会議室

# Reedbush-U概要 OpenFOAM概要

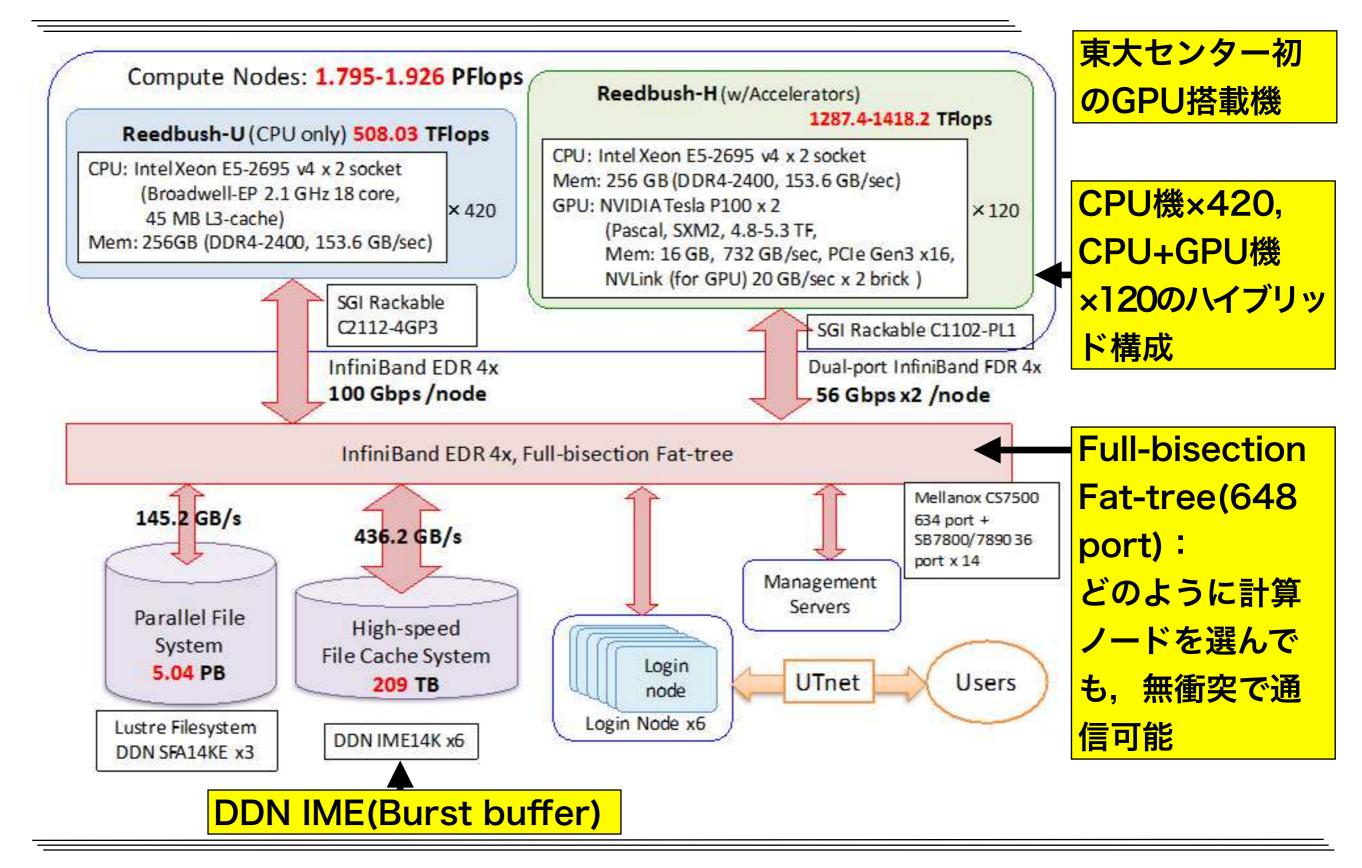
今野 雅 (株式会社OCAEL・東京大学客員研究員)

### 東京大学情報基盤センターの運用システム



2基の大型システム、6年サイクル、現在、5システム運用中

### Reedbushのシステム構成



### Reedbushのハードウェア性能

- CPU(共通): Intel Xeon E5-2695 v4 × 2socket
  - ✓ Broadwell-EP, 2.1GHz, 18core, 45MB L3-cache
- Memory(共通): 256GB(DDR4-2400, 153.6GB/sec)
- ネットワークトポロジー(共通): Full-bisection Fat Tree
- Reedbush-U: CPUクラスタ (2016年9月正式運用開始)
  - ✓ 総理論演算性能(倍精度):508TFLOPS
  - ✓ 総ノード数:420
  - √ インターコネクト: InfiniBand EDR 4x (100 Gbps)
- Reedbush-G: GPUクラスタ (2017年4月正式運転開始)
  - ✓ 総理論演算性能(倍精度): 1.2-1.4 PFLOPS
  - ✓ 総ノード数:120
  - √ GPU : NVIDIA Tesla P100(Pascal) × 2
    - 単体メモリ:16GB、単体メモリ帯域:720GB/sec
    - GPU間接続: NVLink 2 brick(20GB/sec×2)
  - ✓ CPU-GPU接続: PCI Express Gen3 x16レーン(16GB/sec)
  - ✓ インターコネクト: InfiniBand FDR 4x × 2リンク(56 Gbps × 2)



### Reedbushの利用料金

- パーソナル:年間15万円
  - ✓ トークン: 17,280ノード時間(Reedbush-U 2ノード×24時間×360日分)
  - ✓ Reedbush-U:消費係数:1(4ノード以下),2(4ノード超),最大16ノード
  - ✓ Reedbush-H:消費係数:2.5(1ノード),5(1ノード超),最大2ノード
  - √ ディスク: 1TB
- グループ:申込4ノードあたり年間30万円(企業36万円) (N:申込ノード数)
  - ✓ トークン: Reebush-U Nノード×24時間×360日分
  - ✓ Reedbush-U:消費係数:1(Nノード以下),2(N超),最大128ノード
  - ✓ Reedbush-H:消費係数:2.5(N/4ノード以下),5(N/4超え),最大32ノード
  - ✓ ディスク(グループ当り):4TB
- 企業以外はFX10とのトークン相互交換可能
- 企業利用は利用審査があり、簡単な成果公開が必要
- 1週間お試しアカウント付き講習会(年10回程度),トライアルユース(無償・有償)有り

### トライアルユース

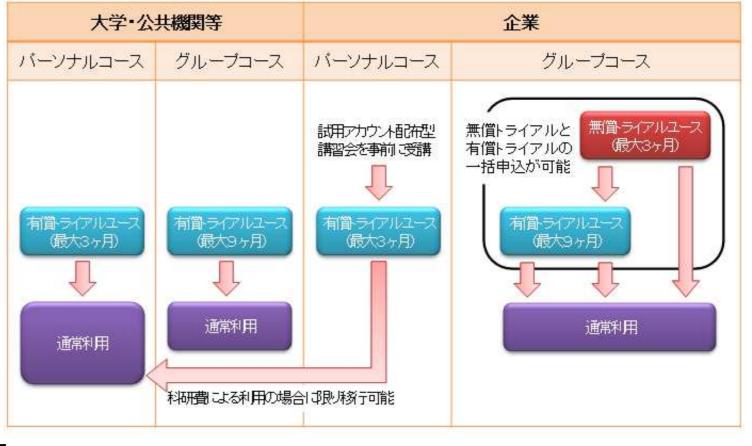
#### 有償トライアルコース(安価)

無償トライアルコース(企業のみ)

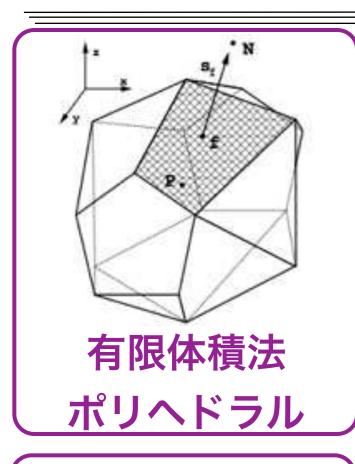
アカデミック利用:最大3ヶ月✓ パーソナルコース✓ グループコース

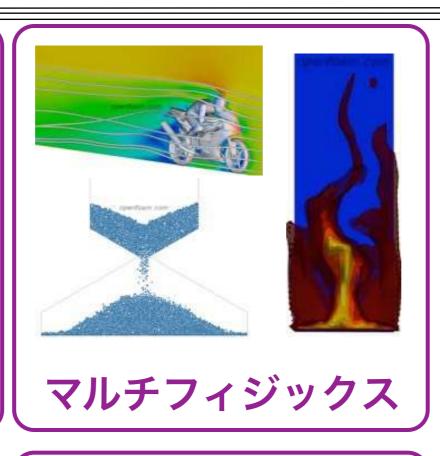


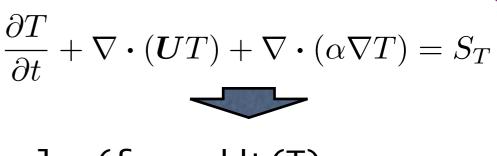
- ✓ パーソナルコース:最大3ヶ月
  - 本講習会の受講が必須、審査無
- ✓ グループコース
  - 無償トライアルユース:最大3ヶ月
  - 有償トライアルユース:最大9ヶ月
  - スーパーコンピュータ利用資格者審査委員会の審査が必要(年2回実施)
- ✓ 双方のコースともに、簡易な利用報告書の提出が必要



# OpenFOAMの概要







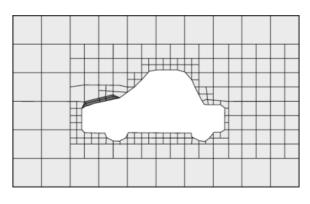
solve(fvm::ddt(T)

+ fvm::div(phi,T)

- fvm::laplacian(DT,T)

== fv0ptions(T));

C++



境界適合Hex メッシャー 乱流モデル:

RAS, LES, DES, ···

線型ソルバー:

AMG, PCG, PBiCG, ···

離散化スキーム: …

多数のモデル実装済

GPL Open Source

カスタマイズ可能 低コストな超並列計算

図出典: [OFF] The OpenFOAM Foundation (http://www.openfoam.org/)

# OpenFOAMの歴史

- 1989年ー2000年:**研究室のFORTRANコード時代**、開発元:英Imperial CollegeのGosman研(Star-CDの開発元)の Henry Weller, Charlie Hill
- 1993年夏:事故により全コード消失。C++で書き直し(FOAM)
- 1999年-2004年:**商用コード期 (FOAM)** Field Operation And Manipulationの略、開発元:▽Nabla(Henry, Hrvoje Jasak, Mattijs Janssensら)、代理店:CAEソリューションズ(元フルイドテクノロジー)
- 2004年12月:オープンソース化 (現在のOpenFOAMに名称変更)、開発元: OpenCFD(Henry, Mattijs, Chris Greenshields)
- 2011年8月15日: SGIによる買収、GPL下のソースの管理や配布は、同時 に設立されたThe OpenFOAM® Foundationが運用
- 2012年9月12日: ESIによる買収、Foundationによる運用は継続
- 2014年: Henryらが独立(CFD Direct社). ソースはFoundationが管理

### 標準ソルバー・チュートリアルのカテゴリ

basic	基礎的なCFDコード
incompressible	非圧縮性流れ
compressible	圧縮性流れ
multiphase	多層流
DNS	直接数値シミュレーション
combustion	燃焼
heatTransfer	熱輸送
electromagnetics	電磁流体
stressAnalysis	固体応力解析
financial	金融工学
lagrangian	ラグランジアン粒子追跡
discreteMethods	Direct Simulation Monte Carlo

### OpenFOAM-v1612+ での総数(概算)

カテゴリ	12
ソルバ	82
チュートリアル	253

#### (参考)上記の概算法

```
$ sol;ls | grep -v doc
|wc -l
```

#### **12**

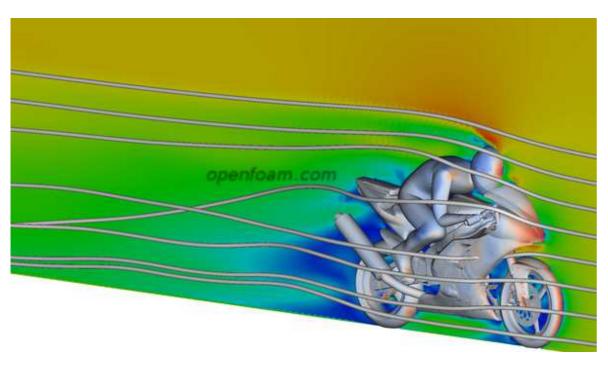
\$ find . -regex '.\*/
Make/files' | xargs
grep EXE | wc -1

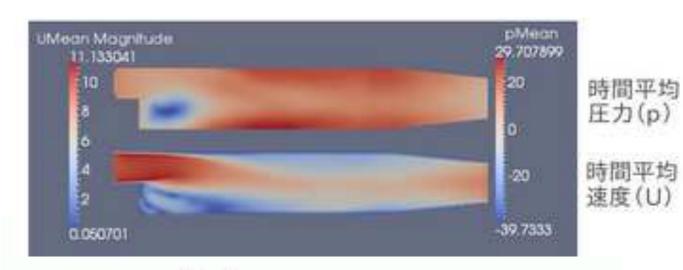
#### 82

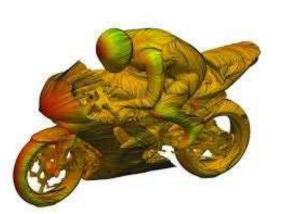
\$ tut;find . -name
controlDict | wc -l

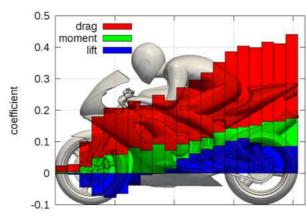
**253** 

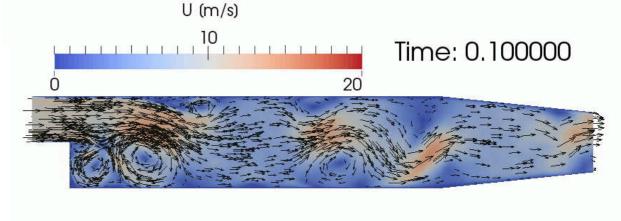
### 単相等温流れのチュートリアル例











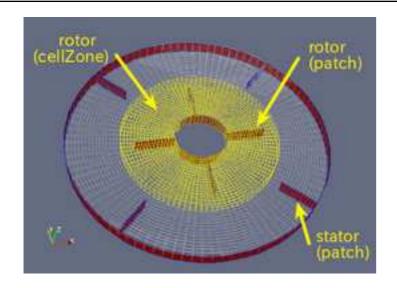
バックステップ流れ(LES) pitzDaily

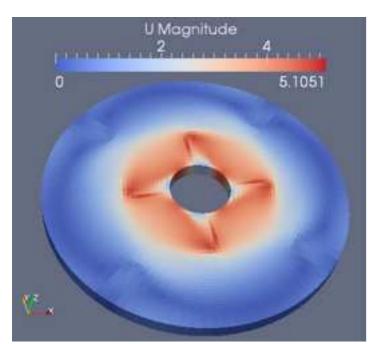
バイク周りの流れ moterBike

図出典: [OFF]

[OFT] オープンCAE勉強会@関西OpenFOAMチュートリアルドキュメント作成プロジェクト( <a href="https://sites.google.com/site/freshtamanegi/">https://sites.google.com/site/freshtamanegi/</a>)

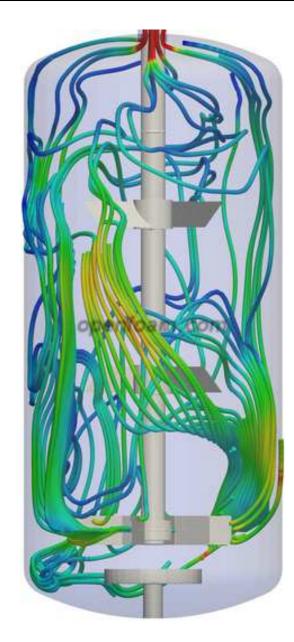
## 回転撹拌槽のチュートリアル例





図出典 [OFT]



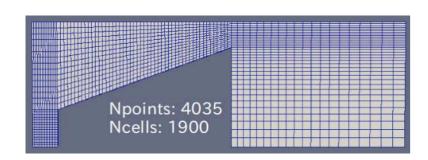


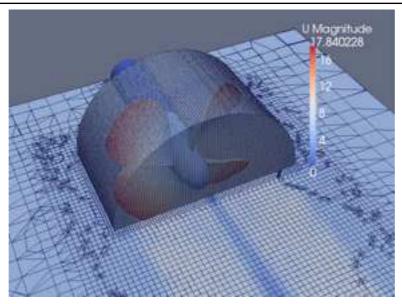
回転攪拌槽の流れ(MRF) mixerVessel2D

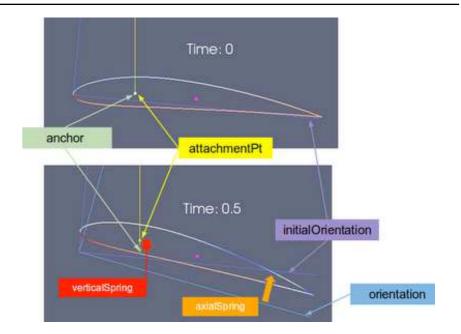
回転攪拌撹拌槽内の流れ mixerVesselAMI

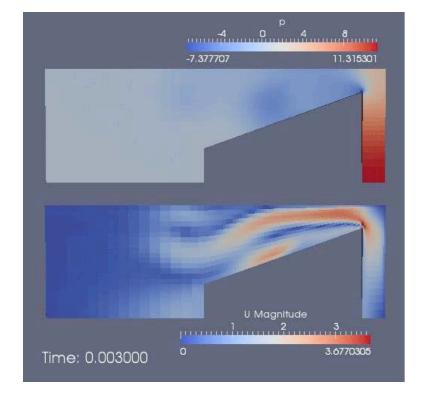
図出典 [OFF]

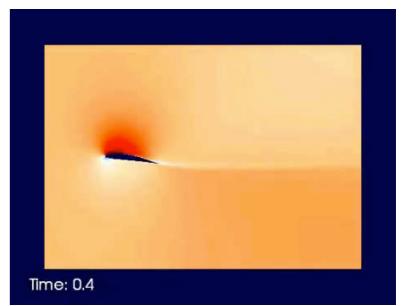
## 移動格子のチュートリアル例









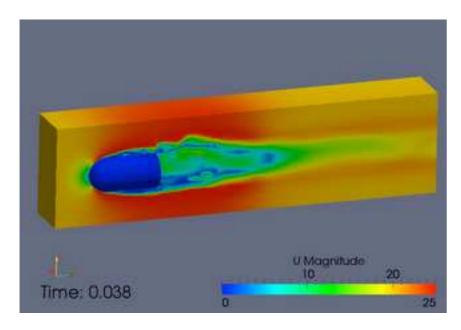


movingCone

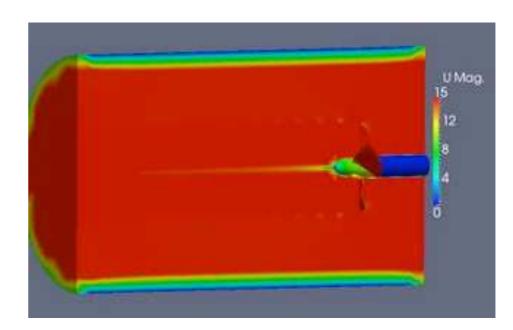
ピストン押し込み流れ スクリューの回転流れ場 propeller 図出典 [OFT]

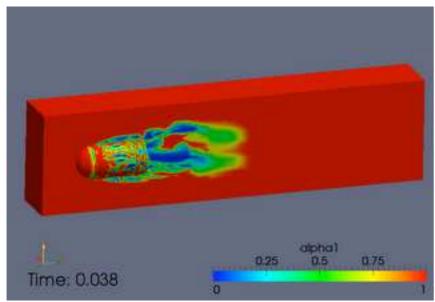
翼型の6自由度剛体運動 wingMotion

### 相変化のチュートリアル例



速度

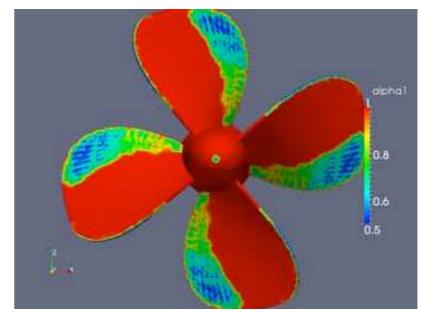




弾丸周りのキャビテーション

cavitatingBullet

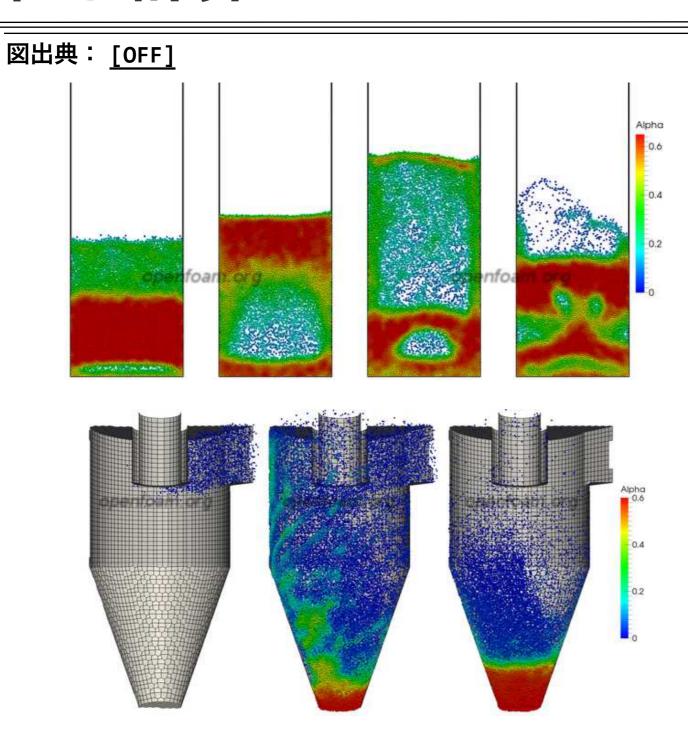
相率



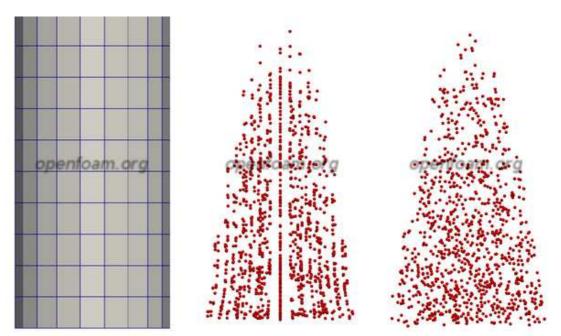
プロペラ周りのキャビテーション propeller

図出典 [OFT]

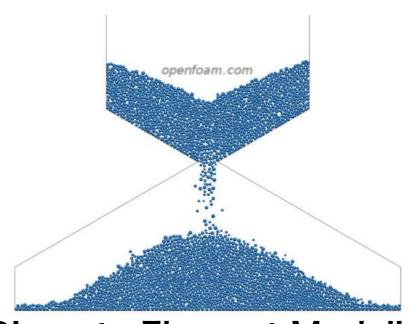
# 粒子計算のチュートリアル例



Multiphase Particle-in-Cell (MP-PIC)

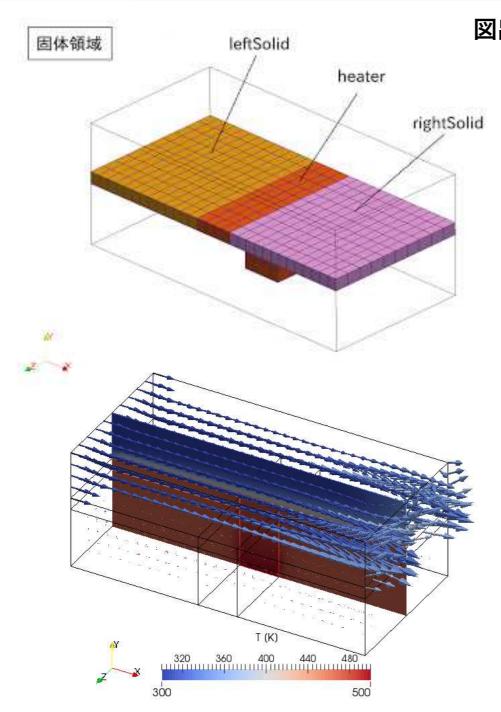


**Particle Tracking** 

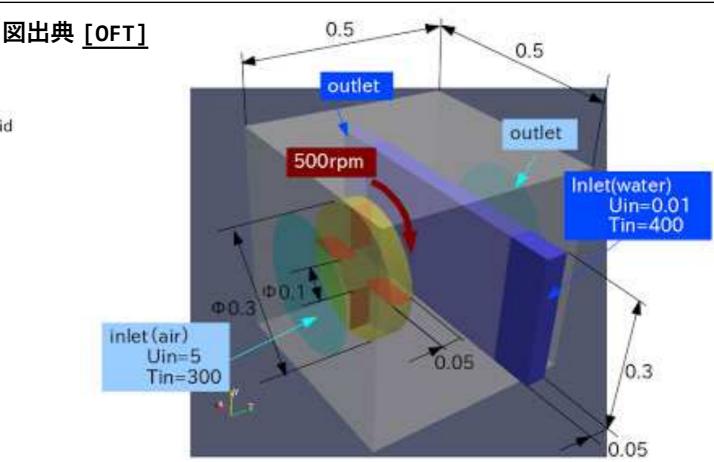


Discrete Element Modeling (DEM)

# 連成熱伝達解析(CHT)のチュートリアル例



連成熱伝達解析 multiRegionHeaterRadiation



回転する熱交換機解析(MRF) heatExchanger

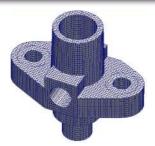
#### OpenFOAMのCHT解析における欠点

- エネルギー保存式を全領域で連成せず、 領域毎に解くので収束が遅い
- 形態係数を用いた放射解析の精度が悪い
- 熱収支計算が容易ではない(熱解析共通)

# OpenFOAMでの代表的な解析手順

#### 前処理(格子生成など)

格子生成 [blockMesh, snappyHexMeshなど]



または

格子生成(サードパーティ) [cfMesh,Salome,gmsh

商用メッシャー等]



必要あれば格子変換 「gmshToFoam等]

#### 解析

初期設定

[setFields等]



領域分割(並列計算時)

[decomposePar]



解析ソルバ

[icoFoam等]



領域統合(並列計算時)

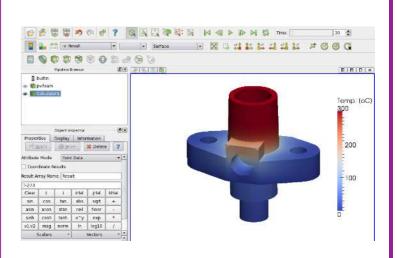
[reconstructPar]

#### 後処理(可視化など)

可視化(サードパーティ)

[ParaView, Visit,

商用可視化ツール等]



様々な結果後処理 [sample等]

# OpenFOAMの稼働環境

Linux, Mac, Windows機で動作





FX10, FX100 (SPARC64機) 東工大 TSUBAME (GPU機あり) FOCUS (Xeon機)



東大 Reedbush-U,H (Xeon機+GPU機)
JCAHPC Oakforest-PACS (Xeon Phi機)
Etc.

Laptop PCクラスタ~100万格子~1000万格子

クラウド スーパーコンピュータ フラッグシップ ~10億格子 ~1000億格子



Amazon EC2(GPU機あり)
Microsoft Azure(GPU機あり)
富士通TCクラウド
Etc. 京(

京 (SPARC64機) RISTがHPCI課題の 京ユーザ向けに最適 化を支援



# OpenFOAM標準のコマンドラインでの解析

#### OpenFOAM標準状態での解析手順:

- ✓ 現在OpenFOAMの標準ではGUI設定ツールが無い
- ✓ テキスト形式の設定ファイルをエディタで修正
- ✓ 前処理、解析、後処理をコマンドラインで実行する
- ✓ サードパーティのGUI可視化ソフトウェアで可視化

#### 利点:

- ✓ スパコンやクラウド等のGUIアプリケーションが動か ないプラットフォームでの実行環境の移植が容易
- ✓ スクリプトによる解析の自動実行を行うのが容易

#### 欠点:

- √ 複雑な解析では、多くのコマンドを打つ必要がある
- ✓ GUIや設定ファイルに対する詳細なマニュアルが無い ので、設定ファイルの中には変更が難しいものが多い

#### 設定ファイル例

```
Lapplication
                   icoFoam;
startFrom
                   startTime;
startTime
                   0;
stopAt
                   endTime;
lendTime
                   0.5;
deltaT
                   0.005;
writeControl
                   timeStep;
¦writeInterval
                   20;
writeFormat
                   ascii;
writePrecision
                   6;
!writeCompression
                   off;
```

#### コマンドライン実行例

```
blockMesh
topoSet

splitMeshRegions -cellZones 略

changeDictionary -region 略
:
faceAgglomerate -region 略
:
viewFactorsGen -region 略
:
chtMultiRegionSimpleFoam
```

### サードパーティGUIツールを用いた解析

#### 解析手順:

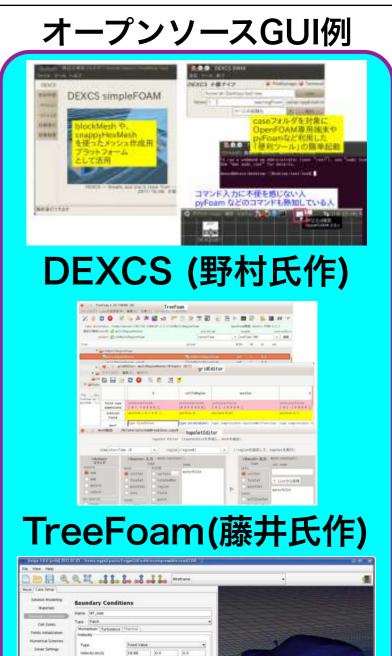
- ✓ GUIツールで基本的な設定を行う
- ✓ 前処理,解析,後処理のコマンドをGUIで起動
- ✓ サードパーティのGUI可視化ソフトウェアで可視化 (コマンドラインでの解析と共通)

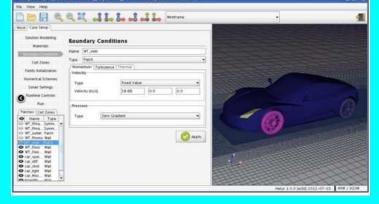
#### 利点:

- ✓ GUIで設定が行えるので、設定作業が容易
- ✓ コマンドを打たずに、基本的な解析が可能

#### 欠点:

✓ GUIツールが対応していないアプリケーションや高度 な設定について、やはりエディタで設定し、コマンド ラインで実行する必要がある





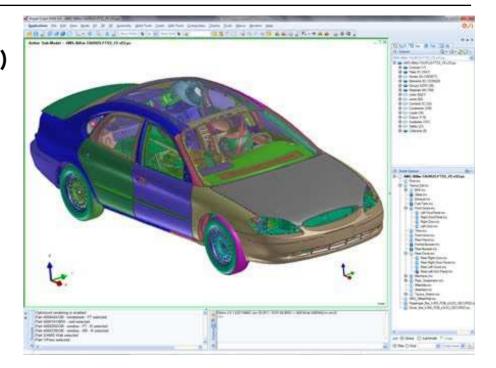
Helyx-OS(Engys)

# OpenFOAMの主な派生(Fork)版

• 商用版

図出典: ESI (<a href="https://www.esi.co.jp/">https://www.esi.co.jp/</a>
news/2014/PressRelease\_0128.html)

- ✓ **HELYX(Engys):** OF拡張版+GUI
- ✓ iconCFD(IDAJ, ICON): OF拡張版+GUI
- √ Visual-CFD(ESI): GUI
- オープンソース版
  - √ HELYX-OS(Engys): GUI



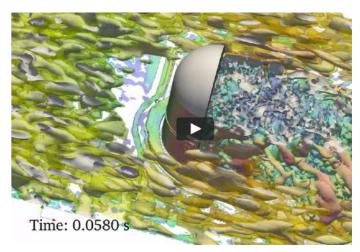
Visual-CFD(ESI)

✓ **foam:** Hrvoje Jasak(クロアチア ザグレブ大学教授, Wikki社 代表)が主導するコミュニティベース版. FSIやBlock coupledソルバ等の公式版に無い機能を実装



foamの 流体・構造連成(FSI)

✓ OpenFOAM+(ESI): 安定化と機能拡張



OpenFOAM+の変動風生成

図出典: www.openfoam.com

# OpenFOAMの派生図

図出典:olaFoam (https://sites.google.com/site/olafoamcfd/)

Imperial College London FOAM

