

RedHat系Linuxへの  
OpenFOAMコンパイル手順まとめ

(OpenFOAM 2.2.x, 2.3.x)

y.imagawa  
14.3.8



# RedHat系へのOpenFOAMインストール

## Download OpenFOAM®

The OpenFOAM Foundation currently distributes OpenFOAM for Linux operating systems in the following ways. The latest release is version 2.3.0, please see [Version 2.3.0 Release Announcement](#) for more information.

### Ubuntu Deb Pack

Version 2.3.0, released 17/02/14

The Ubuntu Deb pack is a good option for a user that wants to install OpenFOAM on a personal computer on which they have Ubuntu Linux running either as the native operating system or within a virtual machine. The pack contains both source code and binaries and includes a list of dependent system software which can be installed automatically to reduce the system administration overhead. It is therefore the **simplest installation** option, making it the ideal choice for those people with no experience of Linux. The pack is updated periodically to incorporate new features and bug fixes.

Go to [Ubuntu Deb Pack Installation](#)

### SuSE RPM Pack

Version 2.3.0, released 17/02/14

OpenFOAM is also released as a source/binary pack for SuSE Linux. It is a little more difficult to install than the Ubuntu pack (above) because not all the dependent software are available as standard SuSE packages, so some additional administration effort is required.

Go to [SuSE RPM Pack Installation](#)

### Red Hat (RHEL) RPM Pack

Version 2.3.0, released 17/02/14

OpenFOAM is also released as a source/binary pack for Red Hat Enterprise Linux 64 bit.

Go to [Red Hat \(RHEL\) RPM Pack Installation](#)

OpenFOAM ver 2.3から、RedHat(RHEL)のパッケージも追加



# OpenFOAMコンパイルのススめ

- 最近、「OpenFOAMをインストールしたい」問い合わせ増加
  - 業務で利用しているLinuxにインストールしたい
  - サーバ購入時に選択できるディストリビューションを使いたい
  - git Repository版を使いたい
- RedHat系のLinuxが多いため、OpenFOAMのコンパイルが必要
- centFOAM?
- そもそも、OpenFOAMのコンパイル方法ってどこにある？



# 想定環境

- OS : CentOS 6.5 (64bit)
- OpenFOAM : ver 2.2.x, 2.3.x
  - ソルバやユーティリティのみ作成
  - paraFoamは作らない (ParaView等は別PCで実行)
- MPI : OpenFOAM ThirdPartyに付いているopenmpiを利用
- gccコンパイラは？



# 使用するコンパイラ

- CentOS標準のg++を使用する
  - コンパイラ導入等の手間が省ける
  - コンパイラが古く、OpenFOAMに使えるか？
- OpenFOAM用のg++を作って利用する
  - ThirdPartyディレクトリに作成する。
  - 最近ではスクリプト一発でコンパイラ作成可



# 使用するコンパイラ

- CentOS標準のg++を使用する
  - コンパイラ導入等の手間が省ける
  - コンパイラが古く、OpenFOAMに使えるか？
- OpenFOAM用のg++を作って利用する
  - ThirdPartyディレクトリに作成する。
  - 最近ではスクリプト一発でコンパイラ作成可



## コンパイルの流れ

1. 必要なrpmパッケージの準備
2. 必要なソースのダウンロード & 配置
3. 設定ファイルの編集
4. コンパイラ、ツールのコンパイル
5. Allwmake
6. テスト (foamInstallationTest)



# 1.必要なrpmパッケージ

CentOSのOSインストール構成が Server の場合、  
以下のパッケージが追加で必要

- gcc
- gcc-c++
- bison
- flex
- m4
- glibc-devel
- glibc-devel.i686
- zlib-devel
- git

管理者権限で以下を実行すると、インストールされる (要外部ネットワーク接続)

```
% yum install gcc gcc-g++ bison flex m4 glibc-devel glibc-devel.i686 zlib-devel
```



## 2. 必要なソースのダウンロード

### OpenFOAM source

`% git clone http://github.com/OpenFOAM/OpenFOAM-2.2.x.git`

`% git clone http://github.com/OpenFOAM/OpenFOAM-2.3.x.git`

### OpenFOAM ThirdParty source

[http://downloads.sourceforge.net/foam/ThirdParty-2.2.2.tgz?use\\_mirror=mesh](http://downloads.sourceforge.net/foam/ThirdParty-2.2.2.tgz?use_mirror=mesh)

[http://downloads.sourceforge.net/foam/ThirdParty-2.3.0.tgz?use\\_mirror=mesh](http://downloads.sourceforge.net/foam/ThirdParty-2.3.0.tgz?use_mirror=mesh)

### OpenFOAM用コンパイラ等

<http://ftp.jaist.ac.jp/pub/GNU/gcc/gcc-4.7.3.tar.bz2>

<http://ftp.jaist.ac.jp/pub/GNU/mpfr/mpfr-3.1.2.tar.bz2>

<http://ftp.jaist.ac.jp/pub/GNU/gmp/gmp-5.1.3.tar.bz2>

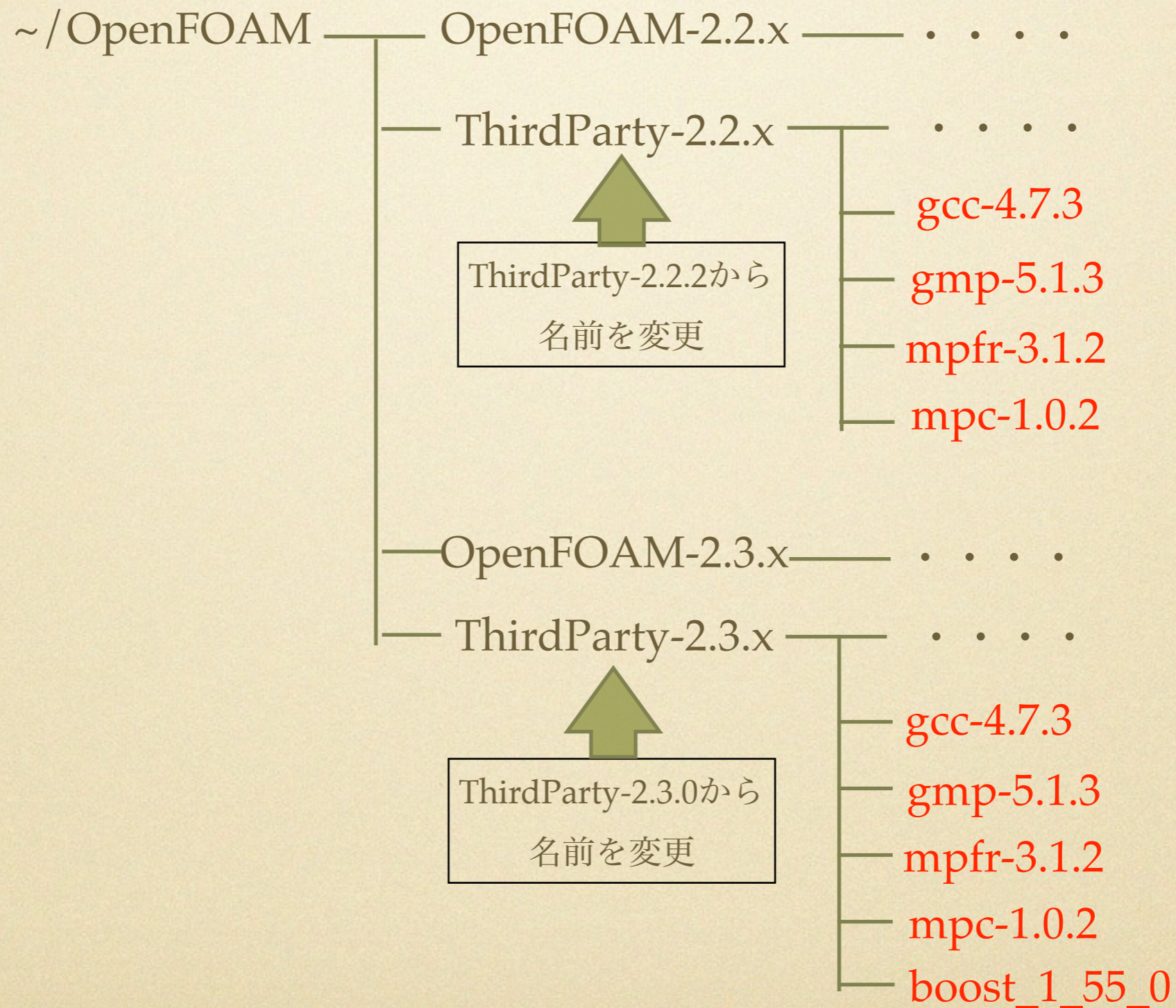
<http://ftp.jaist.ac.jp/pub/GNU/mpc/mpc-1.0.2.tar.gz>

### OpenFOAM ver.2.3用

[http://sourceforge.net/projects/boost/files/boost/1.55.0/boost\\_1\\_55\\_0.tar.bz2/download](http://sourceforge.net/projects/boost/files/boost/1.55.0/boost_1_55_0.tar.bz2/download)



## 2. ファイルの配置





### 3. 設定ファイルの編集

OpenFOAM/ThirdParty-2.2.x/makeGcc47の編集

OpenFOAM/OpenFOAM-2.2.x/etc/bashrcの編集

OpenFOAM/OpenFOAM-2.2.x/etc/config/settings.shの編集

OpenFOAM/ThirdParty-2.2.x/Allwmakeの編集

#OpenFOAM-2.3.x

OpenFOAM/ThirdParty-2.3.x/makeCGALの編集

OpenFOAM/OpenFOAM-2.3.x/etc/config/CGAL.shの編集



## OpenFOAM/ThirdParty-2.2.x/makeGcc47

# Note

# gcc 4.3.2 is known to miscompile GMP 4.3.x on 64-bit machines

#

#-----

gmpPACKAGE=gmp-5.1.3

mpfrPACKAGE=mpfr-3.1.2

mpcPACKAGE=mpc-1.0.2

gccPACKAGE=gcc-4.7.3



OpenFOAM/OpenFOAM-2.2.x/etc/bashrc

#- Compiler location:

# foamCompiler= system | ThirdParty (OpenFOAM)

#foamCompiler=system

**foamCompiler=ThirdParty**

#- Compiler:

# WM\_COMPILER = Gcc | Gcc45 | Gcc46 | Gcc47 | Clang | Icc (Intel icc)

#export WM\_COMPILER=Gcc

**export WM\_COMPILER=Gcc47**

unset WM\_COMPILER\_ARCH WM\_COMPILER\_LIB\_ARCH



OpenFOAM/OpenFOAM-2.2.x/etc/config/settings.sh

```
case "${foamCompiler}" in
OpenFOAM | ThirdParty)
  case "$WM_COMPILER" in
Gcc | Gcc++0x | Gcc46 | Gcc46++0x)
  gcc_version=gcc-4.6.1
  gmp_version=gmp-5.0.4
  mpfr_version=mpfr-3.1.0
  mpc_version=mpc-0.9
  ;;
Gcc47 | Gcc47++0x)
  gcc_version=gcc-4.7.3
  gmp_version=gmp-5.1.3
  mpfr_version=mpfr-3.1.2
  mpc_version=mpc-1.0.2
  ;;
```



## OpenFOAM/ThirdParty-2.2.x/Allwmakeの編集

### openmpiのconfigure修正

(並列実行ができなくなることへの対応。詳細は以下)

<http://mogura7.zenno.info/~et/wordpress/ocse/?p=941>

```
./configure \  
  --prefix=$MPI_ARCH_PATH \  
  --disable-orterun-prefix-by-default \  
  --enable-shared --disable-static \  
#   --libdir=$MPI_ARCH_PATH/lib${WM_COMPILER_LIB_ARCH} \  
  --disable-mpi-f77 \  
  --disable-mpi-f90 \  
  --disable-mpi-cxx \  
  --disable-mpi-profile \  
  $configOpt \  
;
```

↑  
この行をコメントアウト



## OpenFOAM/ThirdParty-2.3.x/makeCGAL

```
# Mixing system and ThirdParty for gmp/mpfr is not supported.  
#  
#-----  
set -x  
  
cgalPACKAGE=CGAL-4.3  
boostPACKAGE=boost_1_55_0  
gmpPACKAGE=gmp-5.1.3  
mpfrPACKAGE=mpfr-3.1.2
```



## OpenFOAM/OpenFOAM-2.3.x/etc/config/CGAL.shの編集

```
# File
#   config/CGAL.sh
#
# Description
#   Setup file for CGAL (& boost) include/libraries.
#   Sourced from OpenFOAM-<VERSION>/etc/bashrc
#-----

boost_version=boost_1_55_0
cgal_version=CGAL-4.3
```



# makeGcc47

## OpenFOAM用のコンパイラを作る

```
% ./~/OpenFOAM/OpenFOAM-2.2.x/etc/bashrc (OpenFOAM環境変数読み込み)
```

(以下のようなWarningが出るが、気にしない)

```
Warning in /home/imagawa/OpenFOAM/OpenFOAM-2.2.x/etc/config/settings.sh:
```

```
Cannot find /home/imagawa/OpenFOAM/ThirdParty-2.2.x/platforms/linux64/gcc-4.7.3 installation.  
Please install this compiler version or if you wish to use the system compiler,  
change the 'foamCompiler' setting to 'system'
```

```
% cd ~/OpenFOAM/ThirdParty-2.2.x
```

```
% ./makeGcc47
```

(コンパイル実行)

```
% exit (OpenFOAM環境から抜ける)
```



# makeCmake

ParaViewをコンパイルするのに必要なcmakeを作成

OpenFOAM ver.2.3では、foamyHexMeshをコンパイルするのに必要

```
% ./~/OpenFOAM/OpenFOAM-2.2.x/etc/bashrc (OpenFOAM環境変数読み込み)
```

(コンパイラ環境ができていれば、Warningは出ない)

```
% cd ~/OpenFOAM/ThirdParty-2.2.x
```

```
% ./makeCmake
```

(コンパイル実行)

```
% exit (OpenFOAM環境から抜ける)
```

makeCmake実行直後は、cmakeにPATHが通ってない

OpenFOAM環境変数を読み直す



# makeCGAL

OpenFOAM ver.2.3で、foamyHexMeshをコンパイルするのに必要

```
% ~/OpenFOAM/OpenFOAM-2.3.x/etc/bashrc (OpenFOAM環境変数読み込み)
```

(コンパイラ環境ができていれば、Warningは出ない)

```
% cd ~/OpenFOAM/ThirdParty-2.3.x
```

```
% ./makeCGAL
```

(コンパイル実行)

```
% exit (OpenFOAM環境から抜ける)
```



# Allwmake

mpi, scotch, OpenFOAM library, applicationをコンパイルする

```
% . ~/OpenFOAM/OpenFOAM-2.1.x/etc/bashrc (OpenFOAM環境変数読み込み)
```

```
% export WM_NCOMPPROCS=4 (4 並列でコンパイル実行)
```

```
% cd ~/OpenFOAM/OpenFOAM-2.2.x
```

```
% ./Allwmake
```

(コンパイル実行)

```
% exit (OpenFOAM環境から抜ける)
```



# テスト

```
% foamInstallationTest
```

```
• • • • •
```

```
Summary
```

---

```
Base configuration ok.
```

```
Critical systems ok.
```

```
Done
```