

構造解析ソフトCalculixのWindows環境 でのbuildの試み

OpenCAE勉強会@岐阜
SH

本日の発表内容

- **背景**

Calculixとは？

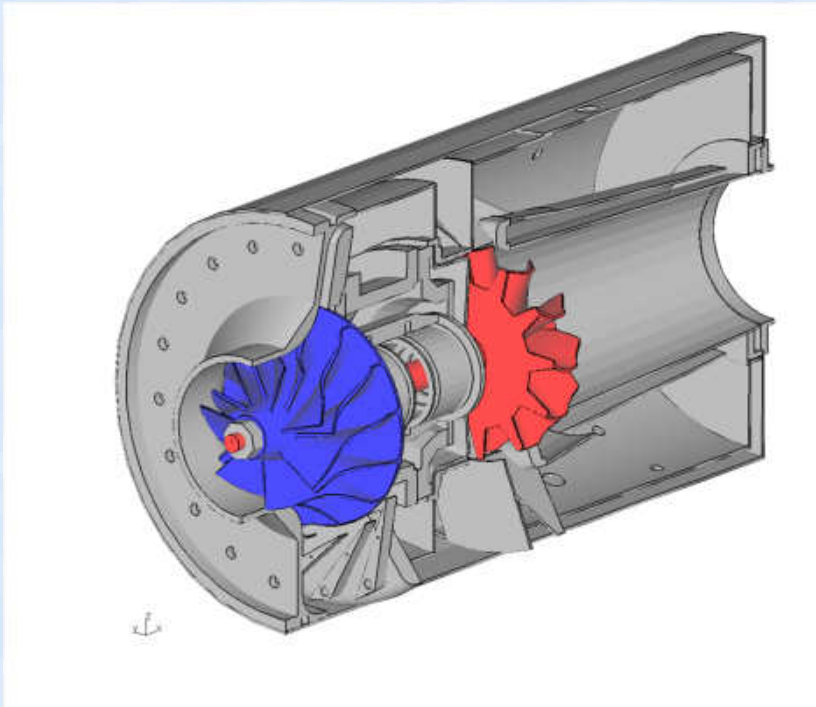
- **Calculixインストール
(Windowsバイナリ編)**
- **Calculixインストール
(Windowsソースから編)**

- **まとめ**

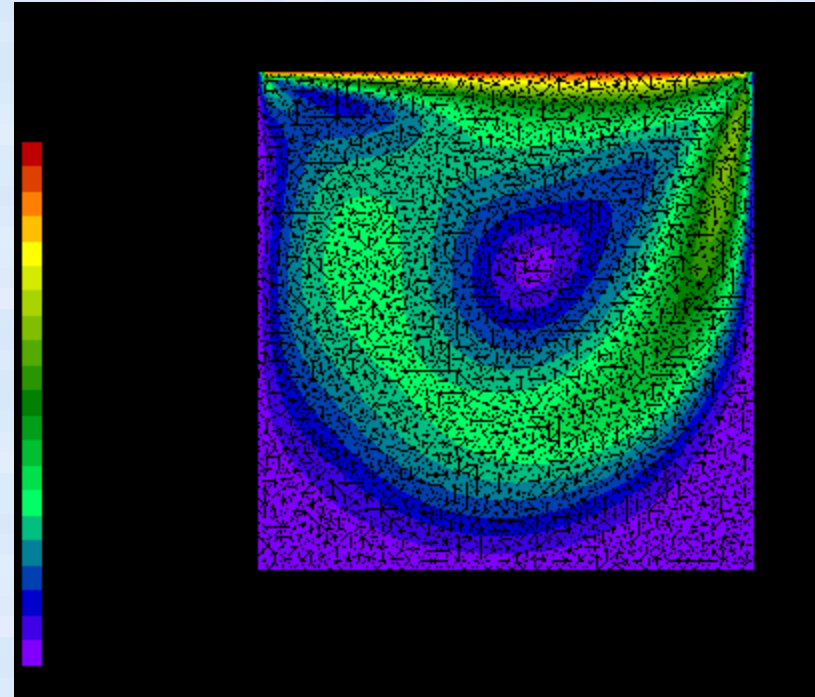
背景

- **CALCULIXソースからの構築方法をすっかり忘れてしまったので、復習のため、WINDOWS上でのビルド方法について確認してみた。**
- **なお本資料は2015/6月の岐阜勉強会の資料“CalculixのLinux(Ubuntu)へのインストール”をベースにしておりますそちら参考にしてください。** <http://opencae.gifu-nct.ac.jp/pukiwiki/index.php?%C2%E8%A3%B4%A3%B0%B2%F3%CA%D9%B6%AF%B2%F1%A1%A7H270613>

Calculixとは?①



Calculix HP から



流体解析の例Cavity FLOW in Calculix

- 商用ソフトABAQUSと同様の入力書式をもつオープンソース ABAQUSを仕事で使っている人は文法を勉強しないでそのまま使える。知らない人もABAQUSのマニュアルを見れば大体使い方が分かる。(テキスト入力ベースのモデラー、メッシュャー、ソルバ、POSTを包含した非線形構造解析ソフト、一部流体解析も可能)
- <http://www.bconverged.com/calculix> にてWindows実行バイナリも公開
- Linux で利用する場合は本家のHP からソースをダウンロードしてコンパイル→ <http://www.dhondt.de/> するかCaelinux(DVD-iso)などのバイナリを利用する。 **ソースからのコンパイルは結構大変。**
- 現在の最新版は**2.13** ソースコードが公開中
- 非線形(大変形、接触解析、材料非線形(塑性、クリープ、温度依存etc)が可能
- 課題:使っている標準行列ソルバ(Spools)→ アウトコアメモリ計算に対応していないのであまり大規模な計算(数10万~100万メッシュ程度限界)には対応できなかった(メモリ搭載量による)。 (→ 他ソルバTAUCSなどを使えばある程度規模の大きな計算ができますが、計算時間はかかる)

Calculixとは?②

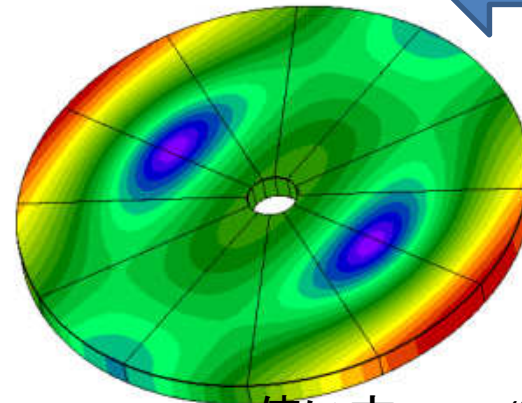
- **Calculixの入力形式はABAQUSと99%同じです（実際は微妙に異なるところがあります）。**
- **ABAQUSの入力ファイルがはき出せるフリーソフトのある方はABAQUS形式ではき出せば、運がよければそのまま動きます。**

```
*HEADING
Model: beam      Date: 10-Mar-1998
*NODE
1, 0.000000, 0.000000, 0.000000
2, 1.000000, 0.000000, 0.000000
3, 1.000000, 1.000000, 0.000000
.
.
.
260, 0.500000, 0.750000, 7.000000
261, 0.500000, 0.500000, 7.500000
*ELEMENT, TYPE=C3D20R, ELSET=Eall
1, 1, 10, 95, 19, 61, 105, 222, 192, 9, 93,
94, 20, 104, 220, 221, 193, 62, 103, 219, 190
2, 10, 2, 13, 95, 105, 34, 134, 222, 11, 12,
96, 93, 106, 133, 223, 220, 103, 33, 132, 219
.
.
.
32, 258, 158, 76, 187, 100, 25, 7, 28, 259, 159,
186, 260, 101, 26, 27, 102, 261, 160, 77, 189
*HSET, HSET=FIX
97, 96, 95, 94, 93, 20, 19, 18, 17, 16, 15,
14, 13, 12, 11, 10, 9, 4, 3, 2, 1
*BOUNDARY
FIX, 1
*BOUNDARY
FIX, 2
*BOUNDARY
FIX, 3
*HSET, HSET=Hall, GENEPATE
1, 261
*MATERIAL, NAME=EL
*ELASTIC
210000.0, .3
*SOLID SECTION, ELSET=Eall, MATERIAL=EL
*HSET, HSET=LOAD
5, 6, 7, 8, 22, 25, 28, 31, 100
**
*STEP
*STATIC
*LOAD
LOAD, 2, 1.
*NODE PRINT, HSET=Hall
```

使い方: `ccx -i "input file"`

Calculixのinputファイルの例

```
DAT25:DISP
Time:1408800.000000
Entity:ALL
max: 9.73e+004
min: 3.29e+003
9.73e+004
9.28e+004
8.83e+004
8.39e+004
7.94e+004
7.49e+004
7.04e+004
6.60e+004
6.15e+004
5.70e+004
5.25e+004
4.81e+004
4.36e+004
3.91e+004
3.46e+004
3.01e+004
2.57e+004
```

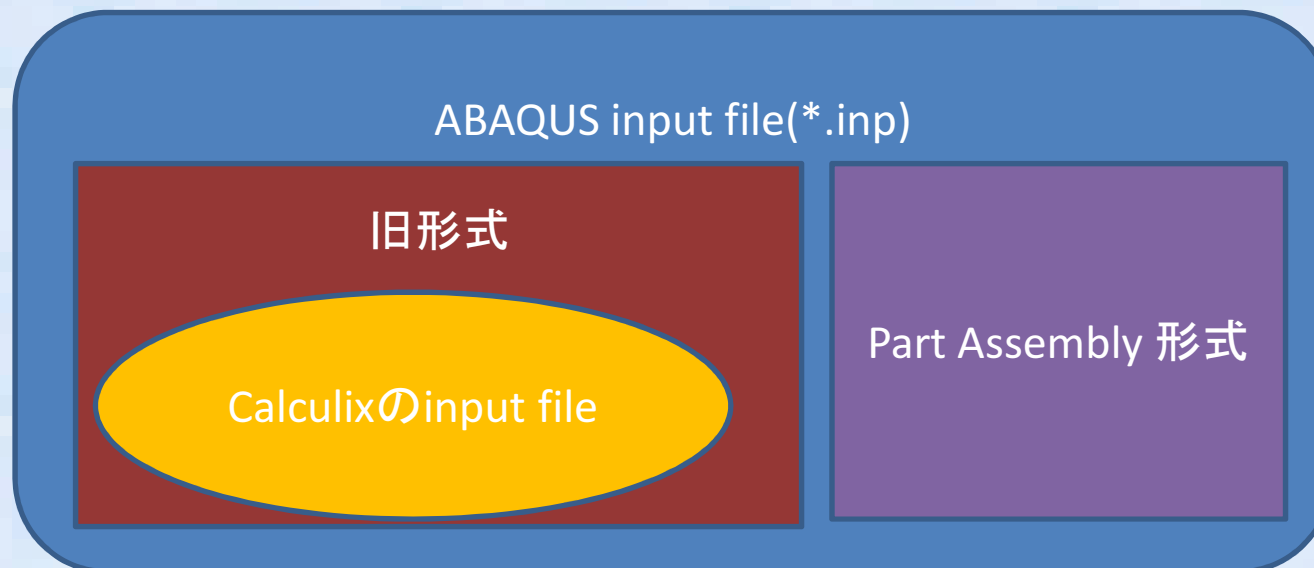


解析結果は専用のポスト処理プログラムで行う (ParaViewを使うことも可能)

使い方: `cgx "結果 .frd file"`

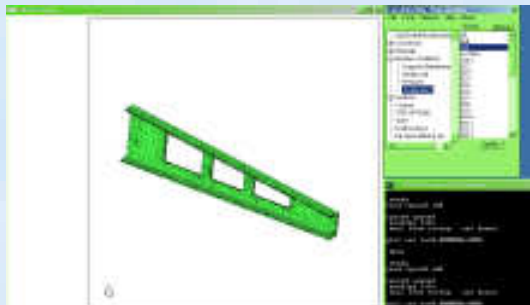
AbaqusとCalculixの比較まとめ

- Calculixの入力ファイル(input)→ほとんどABAQUSでそのまま実行可能
- ABAQUSの入力ファイル(input)→旧形式のファイル出力の一部がCalculixで実行できる。



Calculixインストール(バイナリ編)①

- 本日はExtras版のソースからのビルドを主眼に説明するのでバイナリインストールについてはメモ書き程度ですので、ご了承ください。
- Windows版: 下記の2種類のWindows環境でのバイナリが公開されている。
- ① <http://www.bconverged.com/calculix>
(安定しており良い。商用として有償版を\$55を買うことも可能、有償版はStepなどのCADインターフェースやABAQUSコンバータなどが付属してるらしい?)
→ 有償版は買ったことが無いのでどの程度のものかわかりません!
Windowsインストーラーがついているので、実行するだけでインストールできる(非常に簡単です こちらおすすめ)。
- ② <http://www.calculixforwin.com/>
(cygwinでビルドしたバージョンと思われる?Windowsでのビルド方法も書かれているので、自力でビルドすることも可能と思われる?)
インストールは圧縮されたzipファイルを展開するだけ。V2.6ベース。
Ansysメッシュ形式ファイルのコンバータや独自GUIなどが付属している



CalculixforwinのGUI 通常のGUIの横にメニューが表示され使い易くなっている

Calculixインストール 1

- **CalculixはABAQUSの入力ファイル文法に非常に良く似た書式を持つオープンCAEソフト。**
- **開発元ではソースファイルのみ公開されている。**

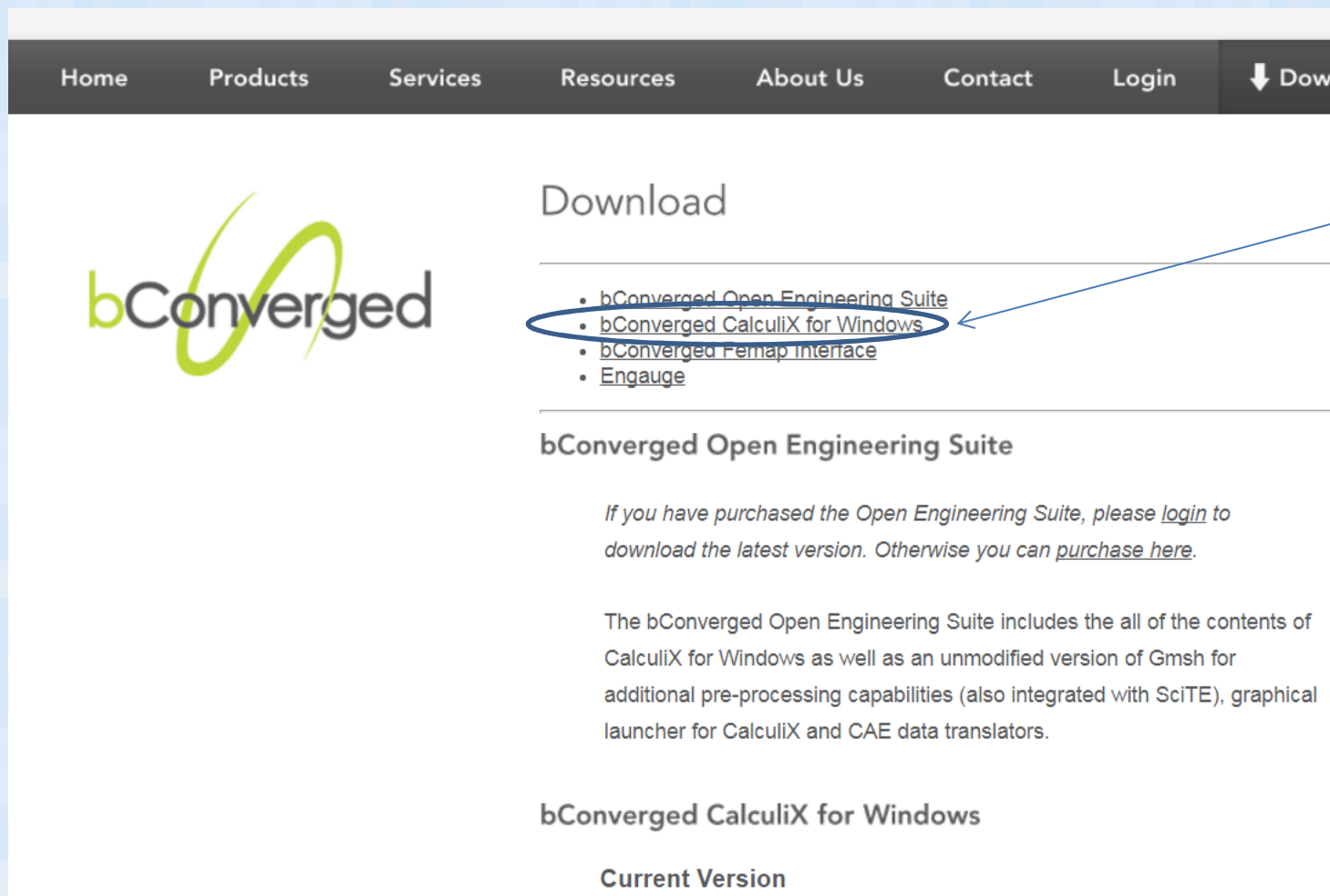
<http://www.calculix.de/>

- **Windows版が2種類公開されているが、今回の実習では下記を利用する**

<http://www.bconverged.com/products.php>

Calculixインストール2

<http://www.bconverged.com/download.php>



Home Products Services Resources About Us Contact Login ↓ Down

Download

- [bConverged Open Engineering Suite](#)
- [bConverged CalculiX for Windows](#)
- [bConverged Femap Interface](#)
- [Engauge](#)

bConverged Open Engineering Suite

If you have purchased the Open Engineering Suite, please [login](#) to download the latest version. Otherwise you can [purchase here](#).

The bConverged Open Engineering Suite includes the all of the contents of CalculiX for Windows as well as an unmodified version of Gmsh for additional pre-processing capabilities (also integrated with SciTE), graphical launcher for CalculiX and CAE data translators.

bConverged CalculiX for Windows

Current Version

選択

Calculixインストール3

選択してダウンロードする

bConverged CalculiX for Windows

Current Version








CalculiX_2_10_win_002.zip CalculiX for Windows 2.10, build 002 (49,362 kb)
Posted 08 May 2016.

This build includes CCX 2.10 (64 bit, multi-thread), CGX 2.10, compiled help file, test cases and two CAD translation tools (VDA and ACIS to FBD). It also contains gnuplot (5.0.3) for plotting results data, ImageMagick (7.0.1-6) for saving images and a custom built SciTE, a text editor which is integrated with the other tools.

This is a bug fix build. The testing script has been updated, env variable CCX_NPROC_STIFFNESS=1 was set to fix multi-threaded issues, all Python code was updated to Python 3.4.

Currently working on a 32bit build. VISTA and earlier Windows operating systems are no longer supported.

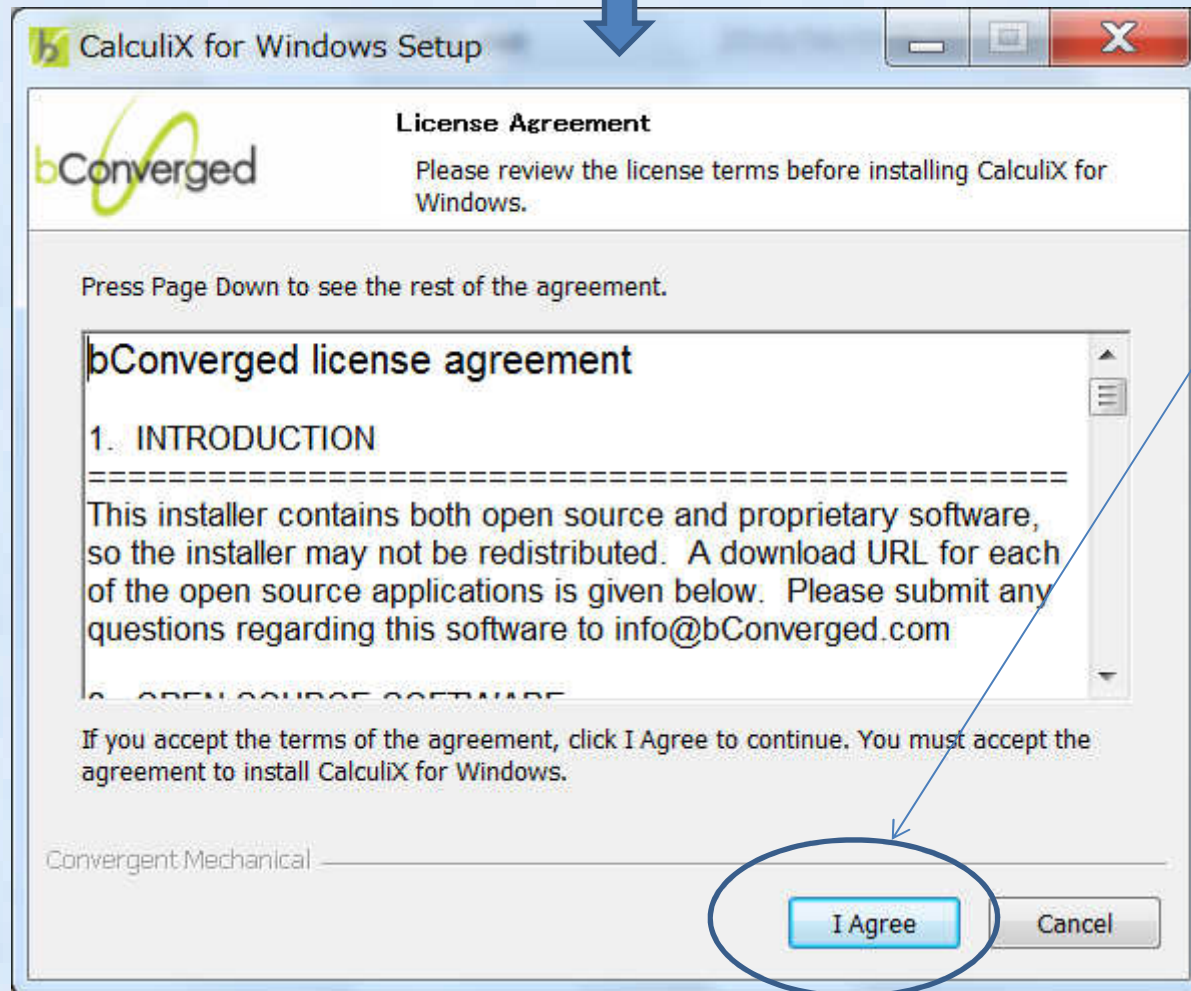
ダウンロードが終わったら
ZIPファイルをインストール
先のフォルダに解凍する。フォル
ダ場所は特に指定は
無く、どこでも良い

 calculixforwin.zip	2013/10/02 16:16	圧縮 (zip 形式) フ...
 CalculiX_2_10_win_002.zip	2017/08/07 12:25	圧縮 (zip 形式) フ...
 CalculiX_2_7_win_003.zip	2015/01/05 12:50	圧縮 (zip 形式) フ...
 CalculiX_2_6.1_win_001.zip	2013/10/02 15:43	圧縮 (zip 形式) フ...
 CalculiX_2_5_win_007.zip	2013/06/23 16:46	圧縮 (zip 形式) フ...
 CalculiX_2_5_win_004.zip	2012/12/20 11:50	圧縮 (zip 形式) フ...
 CalculiX_2_3_win_001.zip	2013/06/30 10:43	圧縮 (zip 形式) フ...

Calculixインストール4

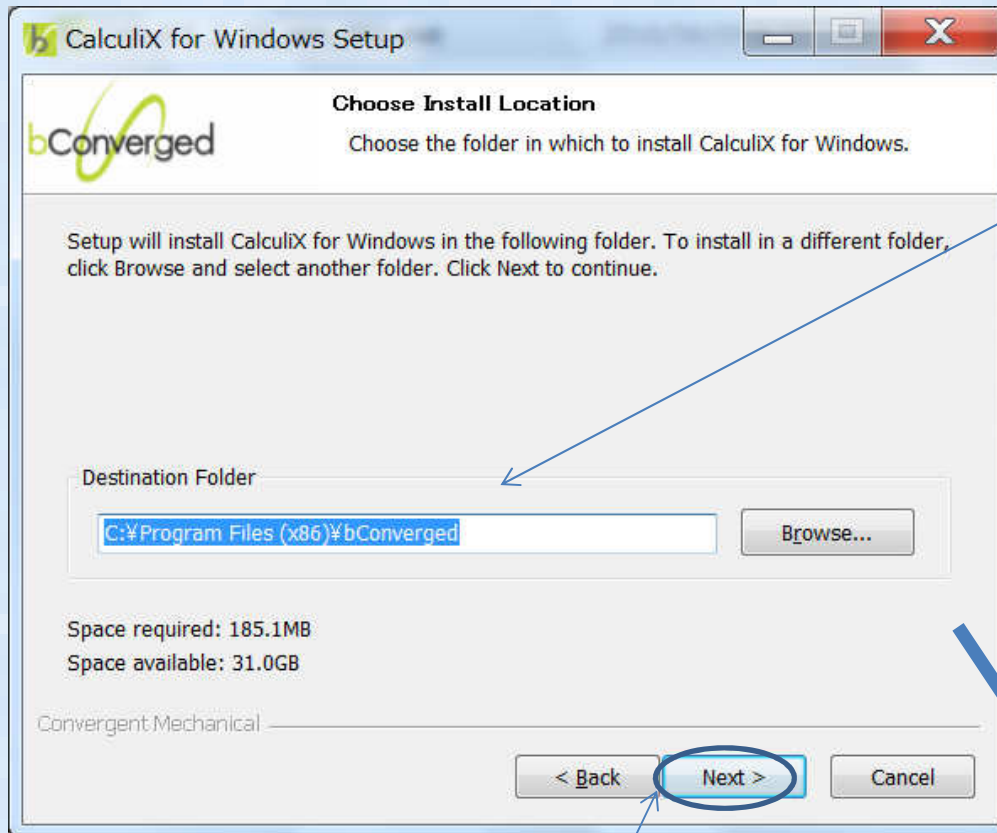
解凍したファイルCalculix_2_10_win_002.exe を実行

 CalculiX_2_10_win_002.exe	2016/06/09 6:22	アプリケーション	49,363 KB
---	-----------------	----------	-----------



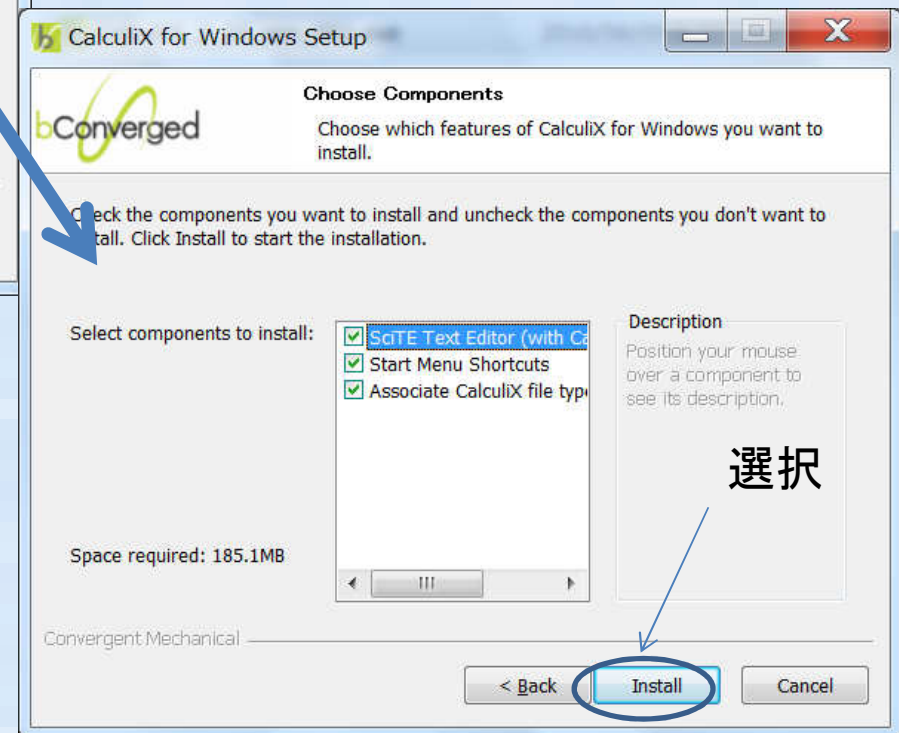
選択

Calculixインストール5



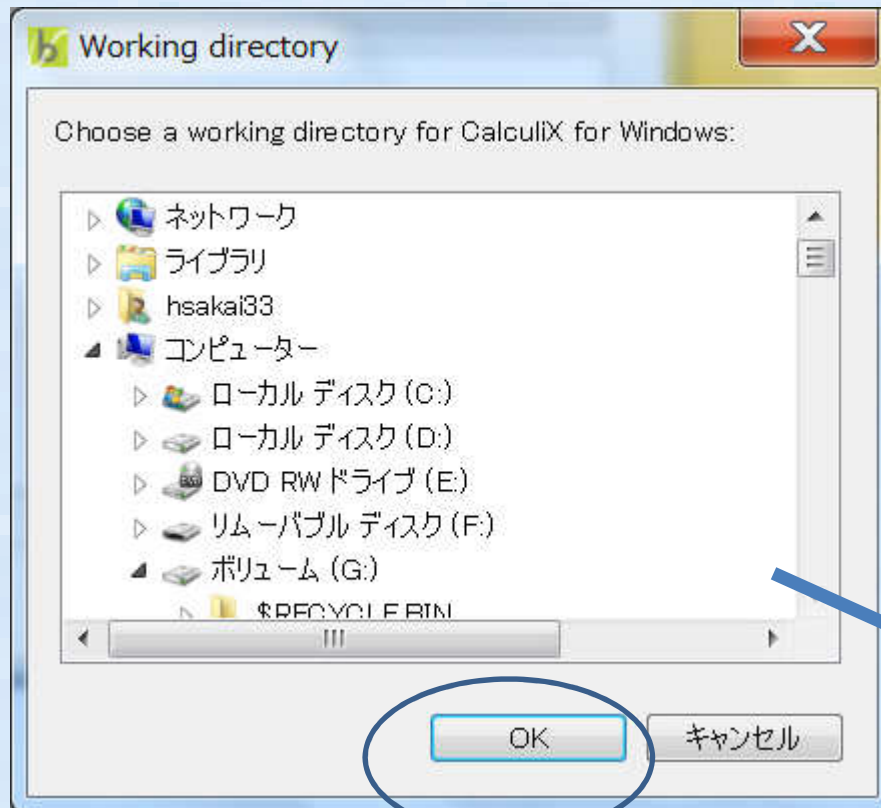
インストール先: 通常はデフォルトで良い

選択

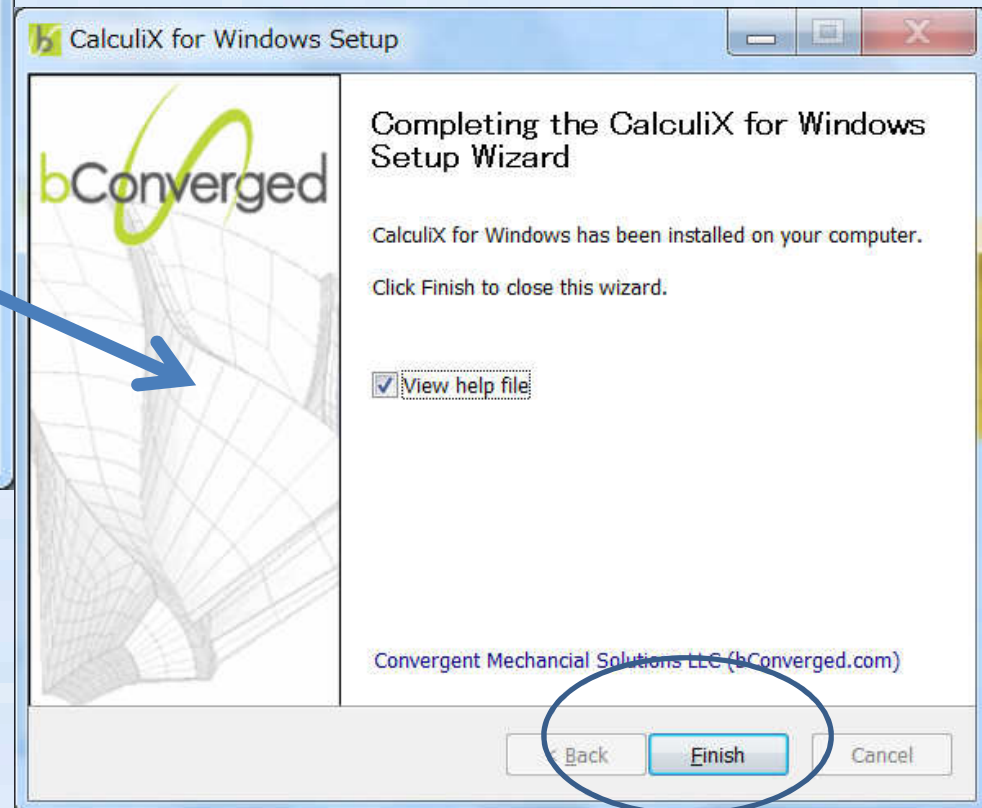


選択

Calculixインストール6



作業ディレクトリを指定する
ABAQUSの作業ディレクトリ
“C:¥Temp”などと同じにして
おくと良いかと思います。

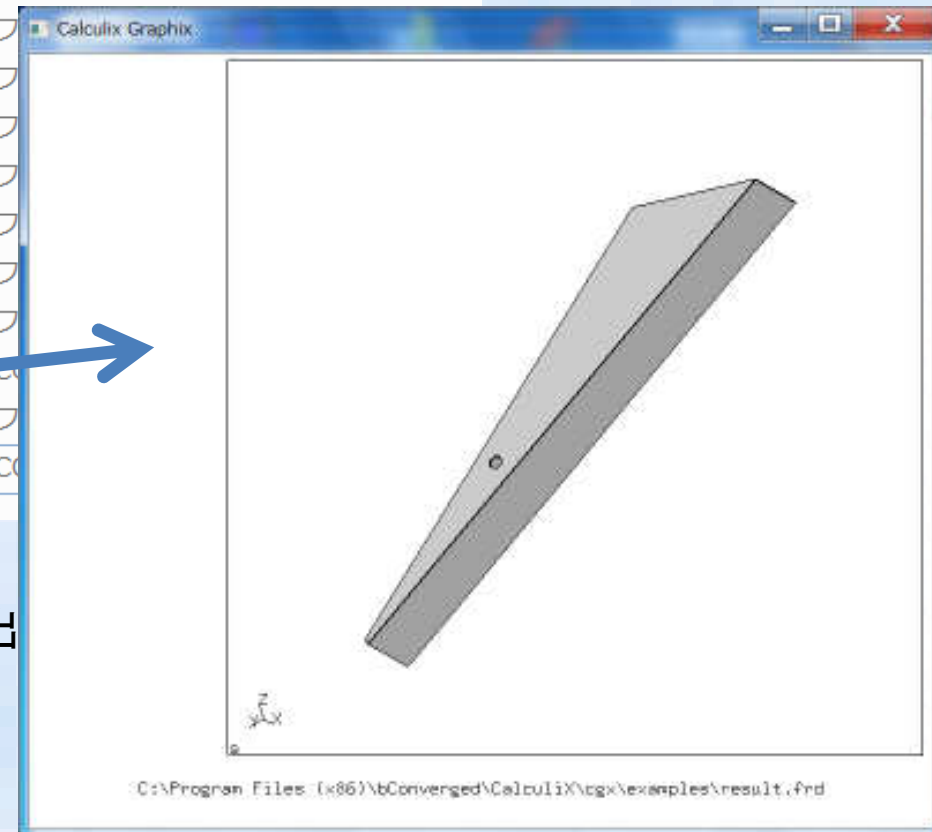
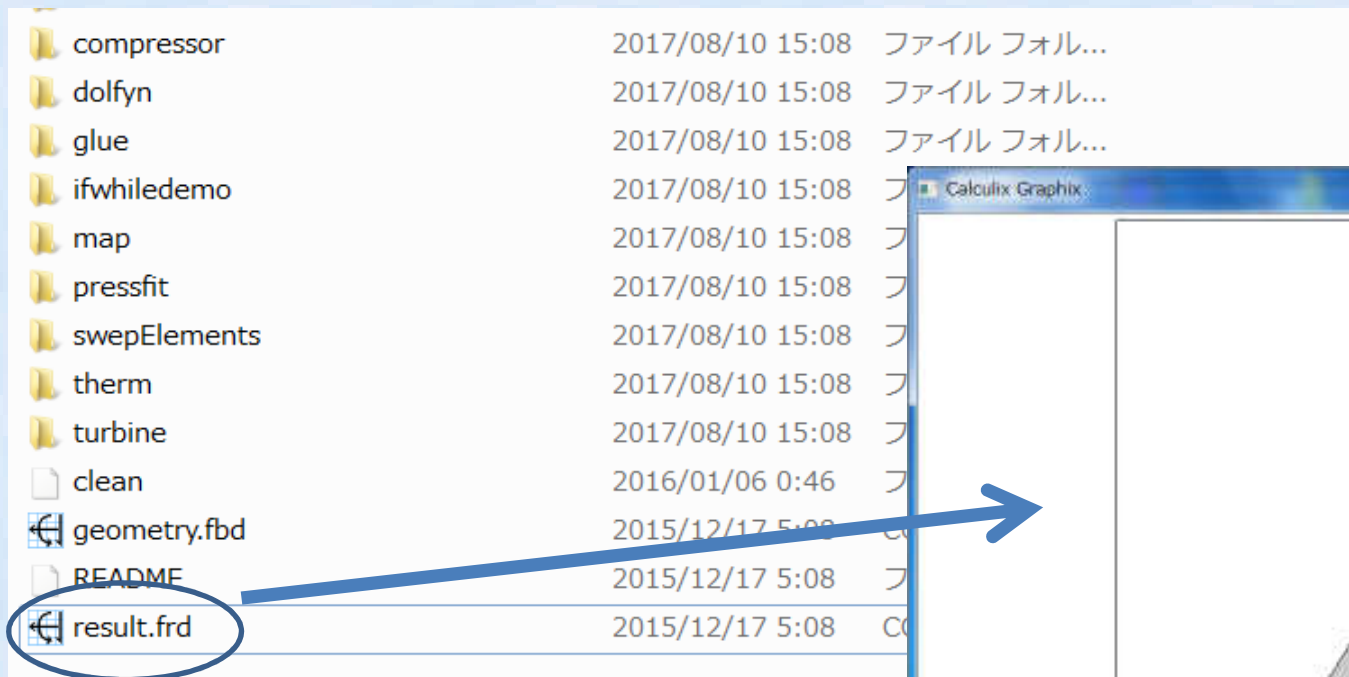


Calculixインストール7

正しくインストールできているか、確認します。

C:\Program Files (x86)\Converged\CalculiX\cgx\examples

(インストール先)にある"result.frd" をエクスプローラからダブルクリック



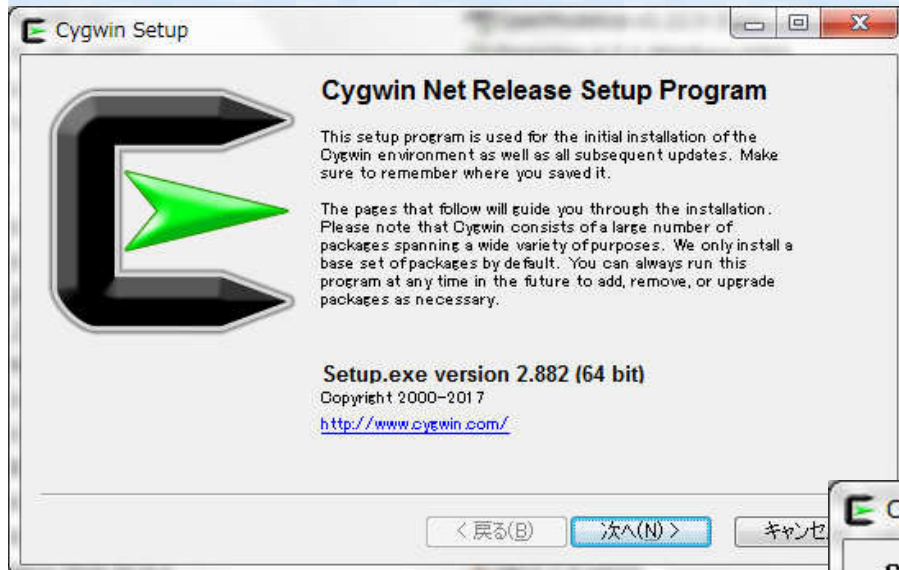
こんなGUI画面が出
てきたらOK

Calculixインストール(Windowsソースから編)①

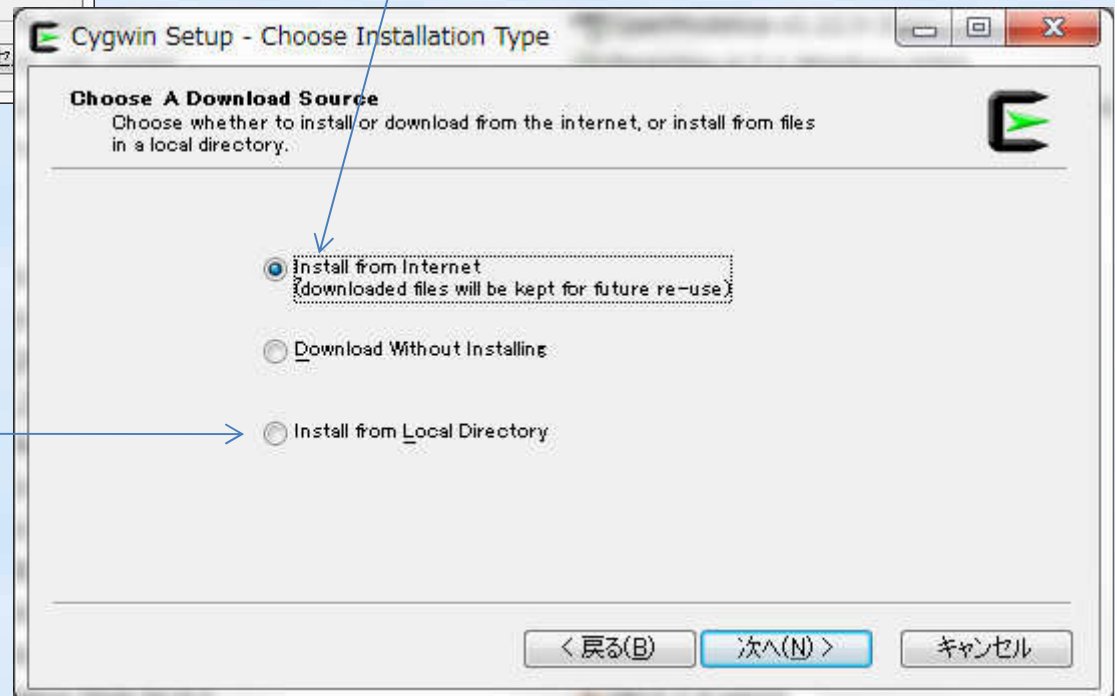
- 基本的にLinuxソースからのビルド方法と同じ。ビルド方法としてVisual-Studio, Intel CompilerまたはLinuxのエミュレーション環境であるcygwinまたはmingwを利用する
- 今回は無料で利用できる環境としてcygwinとmingwでビルドを試みた
↓
- 結果: cygwinでは特に問題なく成功したが、mingwでは失敗する:その後ネットで調べたら色々ソースにパッチを当てる必要があることが分かった(次回に報告するかもしれない?)

Calculixインストール(Windowsソースから編)②

① cygwin のインストール方法 : cygwinのHPからsetup-x86_64.exeをダウンロードして実行する。



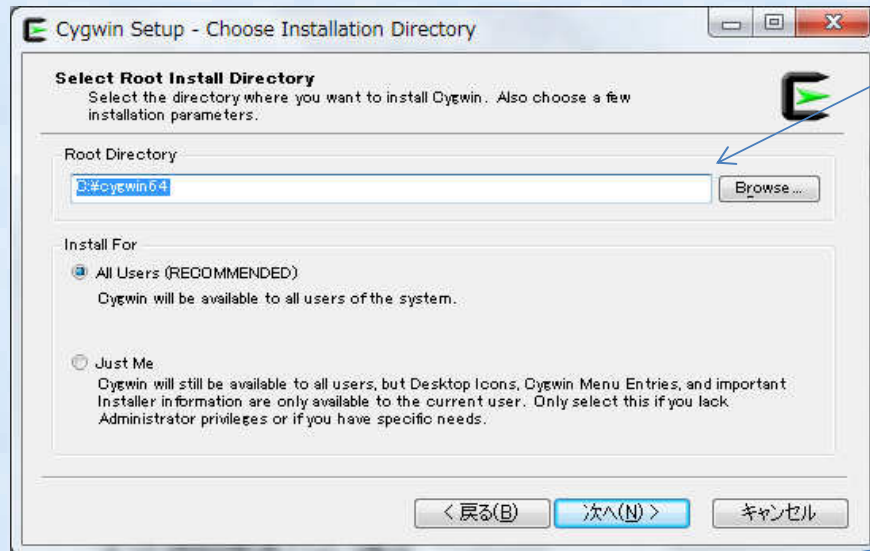
ネット接続環境で通常はこちらを利用する



ダウンロード済みの場合やネット接続環境でない場合はこちらを利用する

Calculixインストール(Windowsソースから編)③

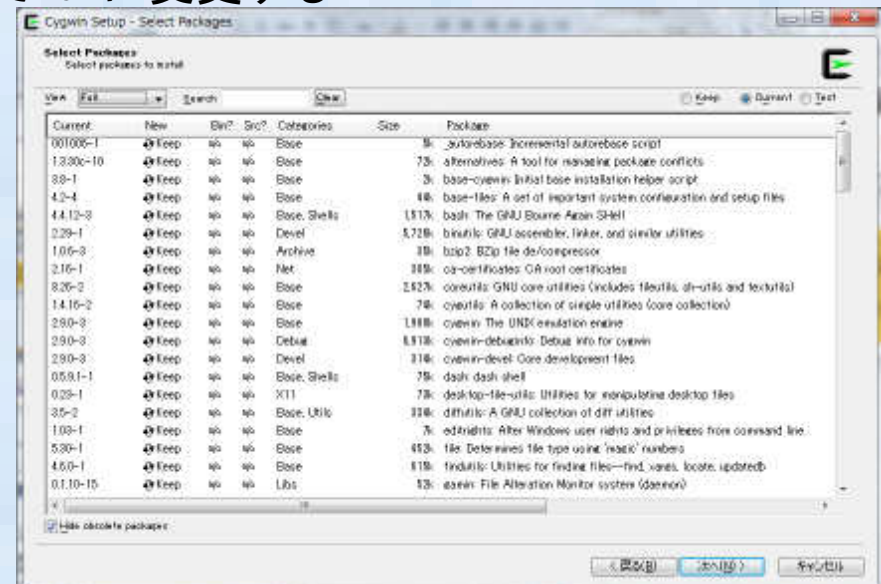
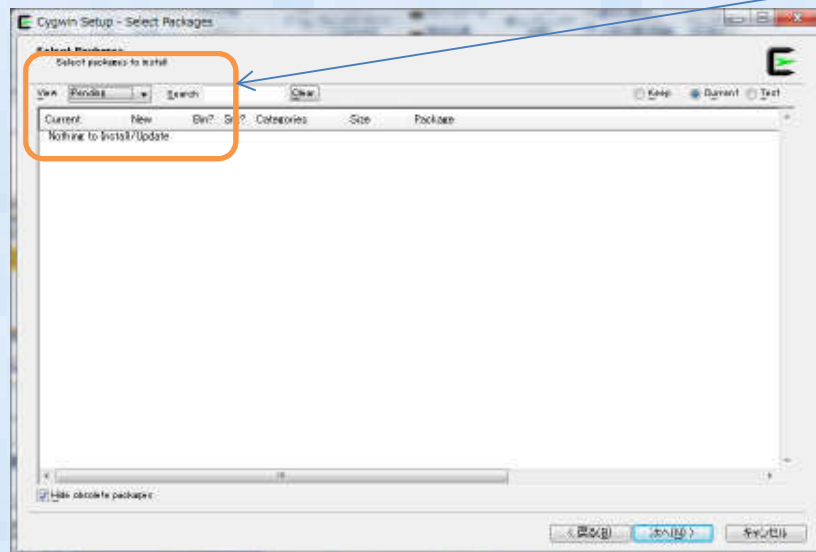
① cygwin のインストール方法 : cygwinのHPからsetup-x86_64.exeをダウンロードして実行する続き。



1) インストール先: デフォルトc:\cygwin64が良い

2) 次にインストールサーバの選択画面になるが、これはどこでも基本的には良いはず? 日本のサーバのどれかを選んでおく

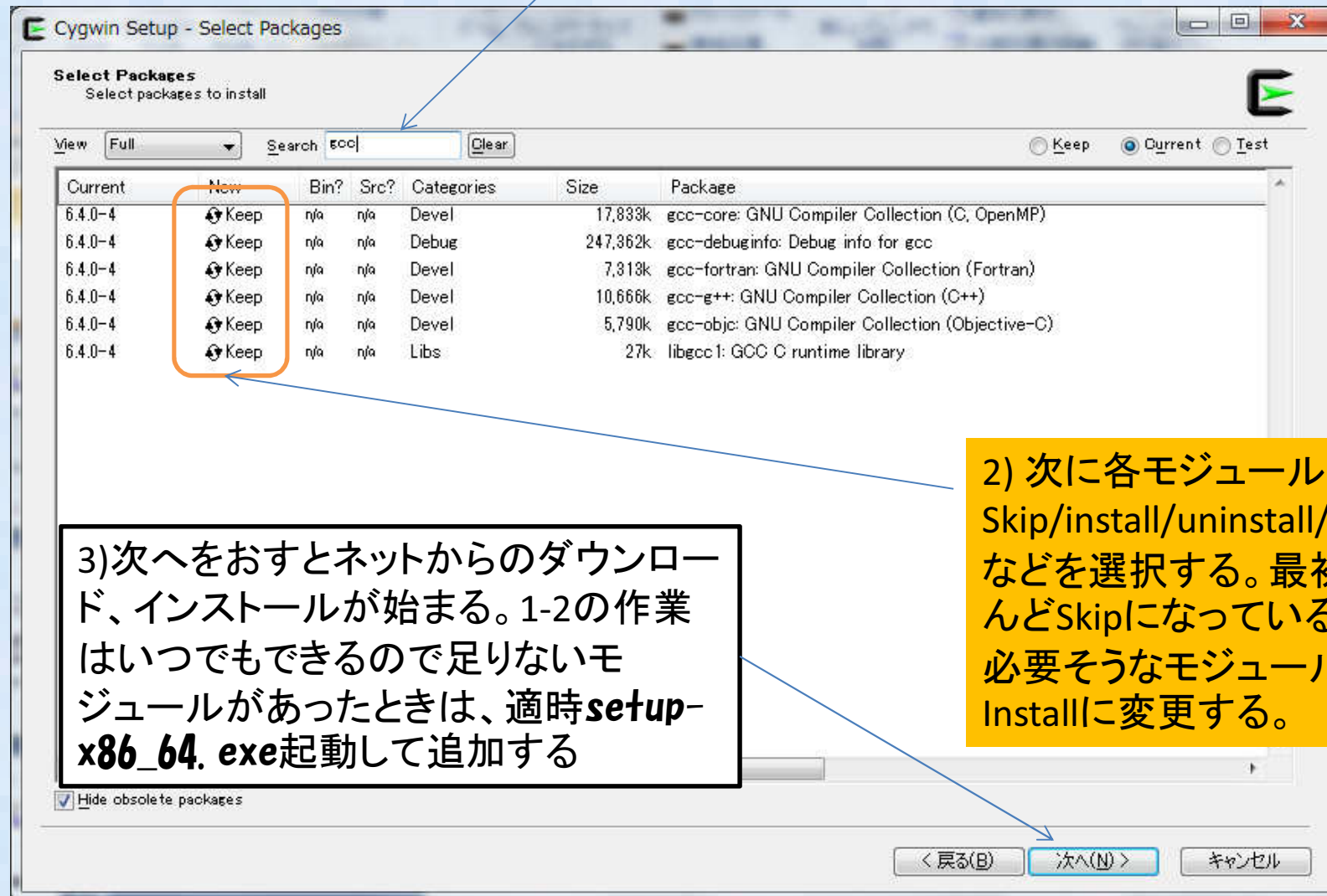
3) デフォルトでPendingになっているのでFullに変更する



Calculixインストール(Windowsソースから編)④

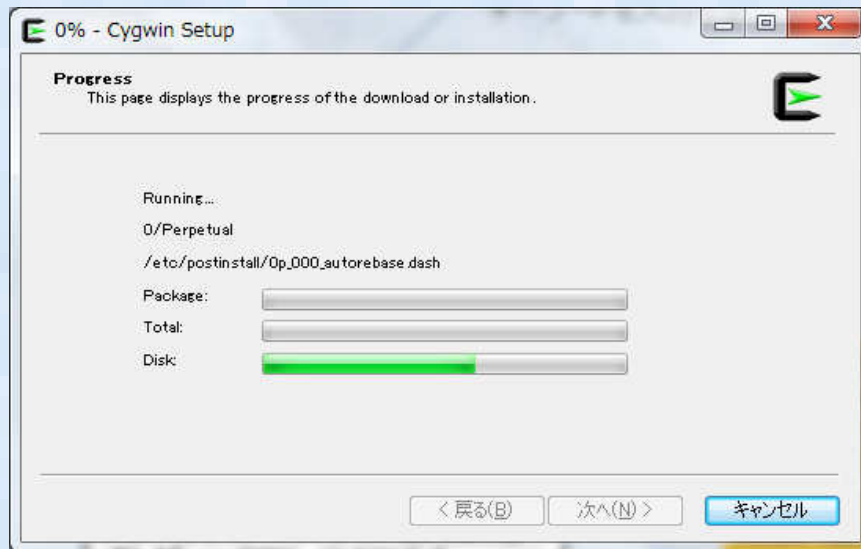
② build環境のインストール : gcc/gfortran/make/perlを選択してサーバからインストールする

1) “gcc”などインストールしたいモジュールのキーワードを入力する

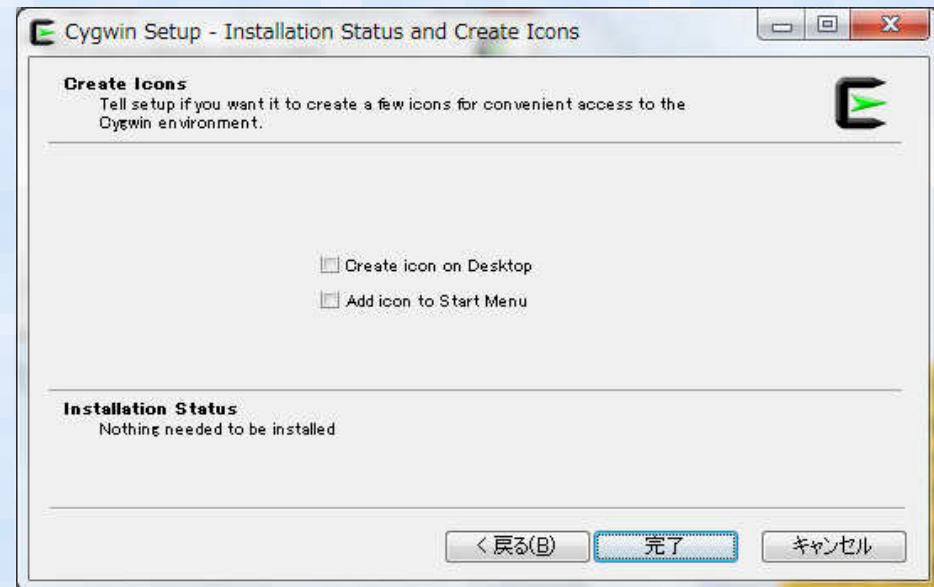


Calculixインストール(Windowsソースから編)④

② build環境のインストール : gcc/gfortran/make/perlを選択してサーバからインストールするの続き



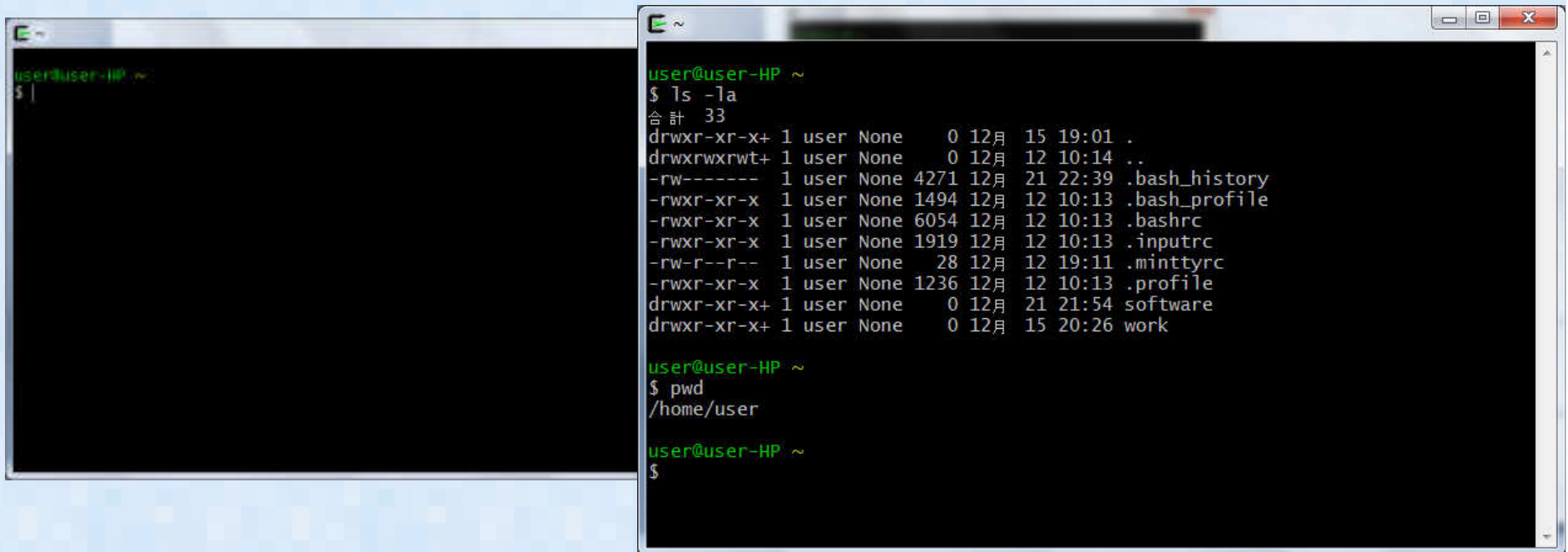
1) “gcc”などのインストールを待つ



2) インストールが終わったら完了、足りないモジュールがある場合は戻るを選択して、再度キーワード入力→インストールを繰り返す

Calculixインストール(Windowsソースから編)⑤

③ cygwin端末画面の起動：cygwin64 Terminal というコマンドが出来るのでこちらを起動して端末画面を起動する。



```
user@user-HP ~
$

user@user-HP ~
$ ls -la
合計 33
drwxr-xr-x+ 1 user None    0 12月 15 19:01 .
drwxrwxrwt+ 1 user None    0 12月 12 10:14 ..
-rw----- 1 user None 4271 12月 21 22:39 .bash_history
-rwxr-xr-x  1 user None 1494 12月 12 10:13 .bash_profile
-rwxr-xr-x  1 user None 6054 12月 12 10:13 .bashrc
-rwxr-xr-x  1 user None 1919 12月 12 10:13 .inputrc
-rw-r--r--  1 user None   28 12月 12 19:11 .minttyrc
-rwxr-xr-x  1 user None 1236 12月 12 10:13 .profile
drwxr-xr-x+ 1 user None    0 12月 21 21:54 software
drwxr-xr-x+ 1 user None    0 12月 15 20:26 work

user@user-HP ~
$ pwd
/home/user

user@user-HP ~
$
```

- あとは基本的にlinuxの端末と同じコマンドが使えるので、問題ないはず？
- linuxコマンドとMS-DOSコマンド両方使える。大文字と小文字の区別がない
- /home/"ログインユーザ名"がデフォルトのディレクトリ
- Windowsのドライブc:¥,d:¥などは/cygdrive/c, /cygdrive/dに対応
- Windowsからはcygwinのインストール先:c:¥cygwin64の直下に/homeなどが見える

Calculixインストール(Windowsソースから編)⑥

④ Calculixのコンパイルその1

基本的にはLinuxの端末画面でコンパイルする場合とまったく同じなので過去の資料(次ページ)を参照ください。

- 1) 事前にspooles(直接法行列ソルバ)とARPACK(固有値ソルバ)をそれぞれビルドしてライブラリを作成しておきます。
- 2) Spoolesはこれまでバグが長期間放置され、そのままではコンパイルできなかったが、最近修正されて、特に問題なくコンパイルできる
- 3) ARPACKはビルドはそのままできるが、1か所修正が必要な個所があり、本修正を行わないとCALCULIX本体のビルド時にエラーになる(UTIL/second.fを一行修正)
- 4) 最後にCalculix本体をビルドします。

```
$ make lib
cd A2 ; make lib
make[1]: ディレクトリ '/cygdrive/c/work3/spooles/A2' に入ります
cd src ; make makeLib
make[2]: ディレクトリ '/cygdrive/c/work3/spooles/A2/src' に入ります
perl ../../makeLib > makeG
/bin/sh: perl: コマンドが見つかりません
make[2]: *** [makefile:25: makeLib] エラー 127
make[2]: ディレクトリ '/cygdrive/c/work3/spooles/A2/src' から出ます
make[1]: *** [makefile:5: lib] エラー 2
make[1]: ディレクトリ '/cygdrive/c/work3/spooles/A2' から出ます
make: *** [makefile:88: lib] エラー 2
```



最初gcc/gfortran/make
だけインストールしたら
perlが無いと怒られた
のでperlを追加した
(calculixのmakefileで
perlを使用している)

Calculixインストール(ソースから編) ①

- extra版では無いケース①-

1: SourceCode圧縮ファイルを下記からDLする。
(最新版以外は公開されていない模様)
<http://www.dhondt.de/> の下の the source code
ccx_2.8p2.src.tar.bz2 をDL これを自分の作業環境に移動
(例えば /home/dexcs の直下にて以下作業を実施する)
\$ bunzip2 ccx_2.8p2.src.tar.bz2 (圧縮解凍)
\$ tar -xvf ccx_2.8p2.src.tar (tar ファイル展開)

(ここから先は必要なライブラリのインストールが必要なためしばらく放置)

2: 必要なライブラリを先にインストールします。

必要なライブラリは直接法行列ソルバのSPOOLESと固有値計算の先に使用するARPACKである。これをソースから先にビルドします。

SPOOLES: <http://netlib.sandia.gov/linalg/spooles/spooles.2.2.tgz> のTGZファイルを

DL とりあえず /home/dexcs/Software/spooles に展開します。

```
$ mkdir /home/dexcs/Software/spooles
```

```
$ mv ./spooles.2.2.tgz /home/dexcs/Software/spooles/
```

```
$ cd /home/dexcs/Software/spooles/
```

```
$ gunzip spooles.2.2.tgz
```

```
$ tar -xvf spooles.2.2.tar
```

この辺の設定はTSUNODAKO さんのブログ↓が詳しいので参考にしてください

<http://freecaetester.blog62.fc2.com/blog-entry-237.html> LINUXにCalculiX ccxをインストールする その1 ダウンロードとSPOOLESのコンパイル

SPOOLESビルド時の
本修正が現在不要に

ここでSpooles ソースにバグなどがあるので、以下2箇所を修正する

① /Tree/src/makeGlobalLib の9行目くらいにdrawTree.c のファイルを呼び出す箇所があるが、このファイルはもう無いのでdraw.c に変更する。変更しないとこの場所でコンパイルエラーで落ちる。

OBJ = Tree

SRC = basics.c ¥

compress.c ¥

draw.c ¥

init.c ¥

② Make.incファイルのC コンパイラの指定を修正

CC = /usr/lang-4.0/bin/cc (大昔のSOLARIS OSの仕様) → CC = /usr/bin/cc または単純に CC = gcc など

修正終わったら make lib を実行 make では無いので注意！！

```
$ make lib
```

これはlinux環境の説明

Calculixインストール(ソースから編) ②

- extra版では無いケース② -

これはlinux環境の説明

- Spooles インストールの続き

make lib が上手くいくと直下に **spooles.a** ができるので、これをCalculix make 時に指定する。

- 並列実行を行うためにはマルチスレッド版のビルドも必要なため、こちらも同時におこなう

spooles のインストールディレクトリ下で

```
$ cd MT
```

```
$ make lib
```

以上でMT/src 下に **spoolesMT.a** ができるのでこれをCalculix make 時利用する

- なお Spooles のビルドは必須では無く、apt-get にてライブラリをインストールすることが可能である。

(ただし、自分でMakeしたものをリンクしないと私の場合はスレッド並列計算がうまくできなかつたので注意！)

spooles をバイナリライブラリから利用する場合は下記にて

```
$ sudo apt-get install libspooles-dev
```

で/usr/libにspooles のライブラリをインストールして Calculix make 時に `-lspooles` のフラグを指定する

(Makefile に `LDFLAGS += -lspooles -lpthread` などの行を追加する)

- 次に固有値計算時に使用するARPACKをビルドします。

Calculixインストール(ソースから編) ③

- extras版では無いケース③ -

- ARPACKインストール: 下記からARPACKのパッケージをダウンロードする。

<http://www.caam.rice.edu/software/ARPACK/>

- arpack96.tar.gz と patch.tar.gz をDLする。

- とりあえず /home/dexcs/Software/ARPACK
に展開します。

```
$ mv arpack96.tar.gz /home/dexcs/Software  
/ARPACK/.
```

```
$ mv patch.tar.gz /home/dexcs/Software  
/ARPACK/.
```

```
$ cd /home/dexcs/Software/ARPACK
```

```
$ gzip -d arpack96.tar.gz
```

```
$ tar -xvf arpack96.tar
```

- パッチファイルの中身を arpack の展開先にそのまま上書き

- ARmake.inc の中身を編集

Fortranコンパイラが **デフォルトで "f77" とか昭和時代**のコンパイラが指定されているので

FC = gfortran に変更

その他下記修正

PLAT = SUN4 → PLAT = INTEL (または PLAT = linux)

FFLAGS = -O -cg89 → FFLAGS = -O2 (または行を削除でも可)

- ARPACK/UTIL/second.f の中身を編集 24行目の下記の行をコメントアウト(または行削除)

*** EXTERNAL ETIME (消しておかないとCalculix make 時にエラーとなる)

- ARPACK のMAKE → \$ make lib これで **libarpack_INTEL.a** などができる。

この辺の設定はTSUNODAKOさんのブログ↓が
詳しいので参考にしてください

<http://freecaetester.blog62.fc2.com/blog-entry-238.html> LINUXにCalculiX ccxをインストールする その2 ARPACKと
ccxのコンパイル

これはlinux環境の説明

Calculixインストール(ソースから編) ④

- extras版では無いケース④ -

- Calculix のソース展開先に戻って本体のmakeを行う
- マルチスレッド並列版のMakefileのテンプレート(Makefile_MT)があるのでこれをベースに自分の環境用に編集する。基本的には先ほどのSpoolesとARPACKの置き場所だけちゃんと書いてあれば問題無いはず？

```
$ cd /home/dexcs/Calculix/ccx2.8p2/src
```

```
$ cp Makefile_MT Makefile
```

これはlinux環境の説明

- Makefile の編集 大体下記のように編集する。終わったらMakeして完了！

```
CFLAGS = -Wall -O3 -I /home/dexcs/Software/spooles -DARCH="Linux" -DSPOOLES -DARPACK -DMATRIXSTORAGE -DUSE_MT=1
FFLAGS = -Wall -O3
CC=cc
FC=gfortran
.c.o :
    $(CC) $(CFLAGS) -c $<
.f.o :
    $(FC) $(FFLAGS) -c $<
include Makefile.inc
SCCXMAIN = ccx_2.8p2.c
OCCXF = $(SCCXF:.f=.o)
OCCXC = $(SCCXC:.c=.o)
OCCXMAIN = $(SCCXMAIN:.c=.o)
DIR=/home/dexcs/Software/spooles
LIBS = ¥
    $(DIR)/MT/src/spoolesMT.a ¥
    $(DIR)/spooles.a ¥
    /home/dexcs/Software/ARPACK/libarpack_INTEL.a ¥
    -lpthread -lm
ccx_2.8p2_MT: $(OCCXMAIN) ccx_2.8p2_MT.a $(LIBS)
    ./date.pl; $(CC) $(CFLAGS) -c ccx_2.8p2.c; $(FC) -Wall -O3 -o $@ $(OCCXMAIN) ccx_2.8p2_MT.a $(LIBS)
ccx_2.8p2_MT.a: $(OCCXF) $(OCCXC)
    ar vr $@ $?
```

赤字部分は書き直し箇所

Cygwin上のMakefileの例

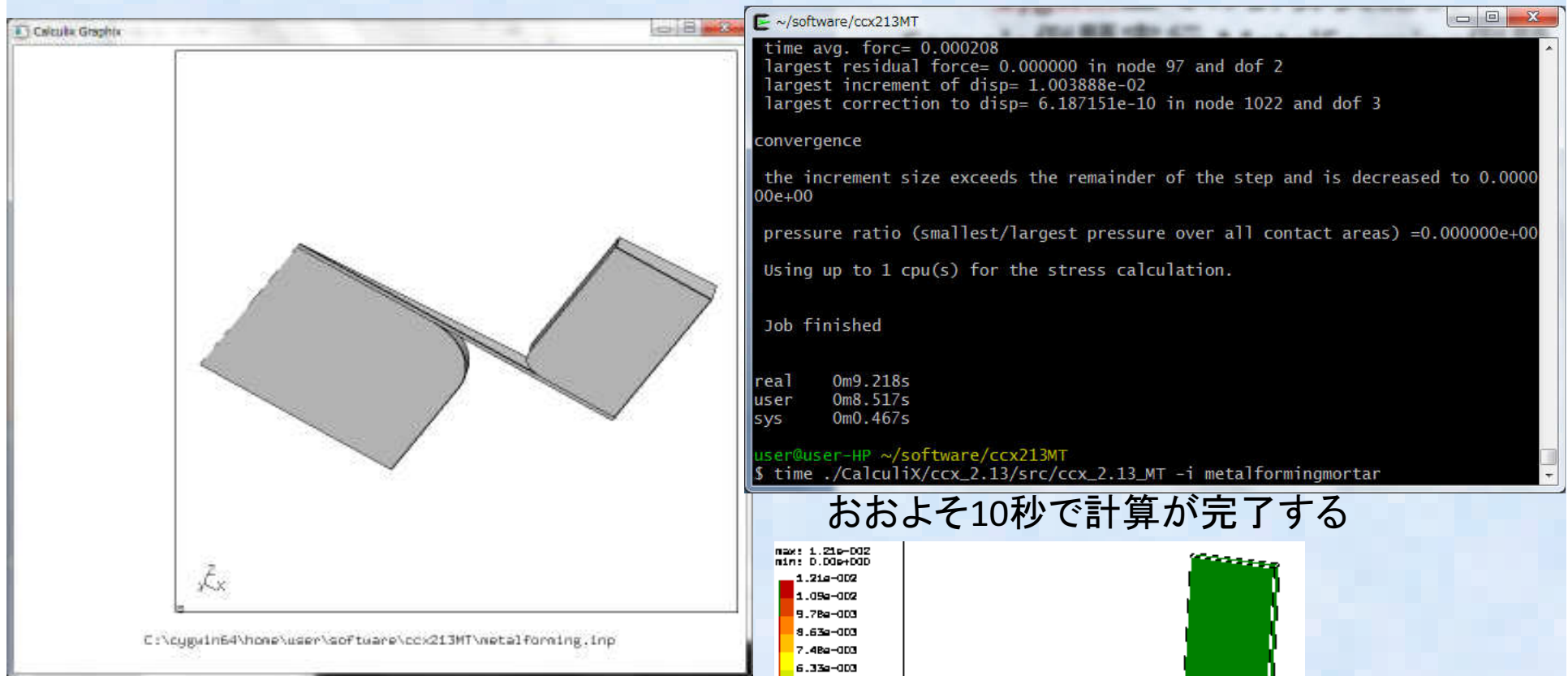
- シングル版, MT(マルチスレッド版)それぞれ下記

```
Makefile - Notepad2
File Edit View Settings 2
1
2 CFLAGS = -Wall -O3 -fopenmp -I /home/user/software/spooles -DARCH="Linux" -DSPOOL
3 FFLAGS = -Wall -O3 -fopenmp
4
5 CC=cc
6 FC=gfortran
7
8 .c.o :
9 $(CC) $(CFLAGS) -c $<
10 .f.o :
11 $(FC) $(FFLAGS) -c $<
12
13 include Makefile.inc
14
15 SCCXMAIN = ccx_2.13.c
16
17 OCCXF = $(SCCXF:.f=.o)
18 OCCXC = $(SCCXC:.c=.o)
19 OCCXMAIN = $(SCCXMAIN:.c=.o)
20
21 DIR=/home/user/software/spooles
22
23 LIBS = \
24     $(DIR)/spooles.a \
25     /home/user/software/ARPACK/libarpack_INTEL.a \
26     -lpthread -lm -lc
27
28 ccx_2.13: $(OCCXMAIN) ccx_2.13.a $(LIBS)
29 ./date.pl; $(CC) $(CFLAGS) -c ccx_2.13.c; $(FC) -fopenmp -Wall -O3 -o $@ $(OCCX
30
31 ccx_2.13.a: $(OCCXF) $(OCCXC)
32 ar vr $@ $?
33
Ln 30 : 33 Col 1 : 79 Sel 0 686 バイト ANSI LF INS Makefiles
```

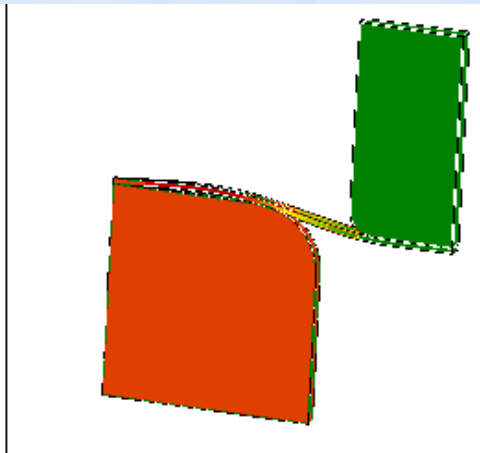
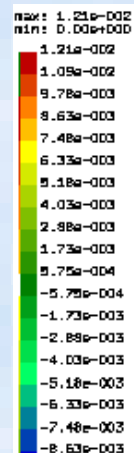
```
Makefile - Notepad2
File Edit View Settings 2
1
2 <CFLAGS = -Wall -O3 -fopenmp -I ../..../spooles -DARCH="Linux" -DSPOOLES -DARPACK -DMATRIXSTORAGE -DNUSE_MT=1
3 FFLAGS = -Wall -O3 -fopenmp
4
5 CC=cc
6 FC=gfortran
7
8 .c.o :
9 $(CC) $(CFLAGS) -c $<
10
11 .f.o :
12 $(FC) $(FFLAGS) -c $<
13
14 include Makefile.inc
15
16 SCCXMAIN = ccx_2.13.c
17
18 OCCXF = $(SCCXF:.f=.o)
19 OCCXC = $(SCCXC:.c=.o)
20 OCCXMAIN = $(SCCXMAIN:.c=.o)
21
22 DIR=../..../spooles
23
24 LIBS =
25     $(DIR)/MT/src/spoolesMT.a \
26     $(DIR)/spooles.a \
27     ../..../ARPACK/libarpack_INTEL.a \
28     -lpthread -lm
29
30 ccx_2.13_MT: $(OCCXMAIN) ccx_2.13_MT.a $(LIBS)
31 ./date.pl; $(CC) $(CFLAGS) -c ccx_2.13.c; $(FC) -fopenmp -Wall -O3 -o $@ $(OCCXMAIN) ccx_2.13_MT.a $(LIBS)
32
33 ccx_2.13_MT.a: $(OCCXF) $(OCCXC)
34 ar vr $@ $?
35
Ln 1 : 35 Col 1 : 79 Sel 0 791 バイト ANSI LF INS Makefiles
```

Cygwin上での計算実施例

Sample例題実行 MetalForming例題



おおよそ10秒で計算が完了する



Cygwin上での計算実施例

Sample例題実行 MetalForming例題

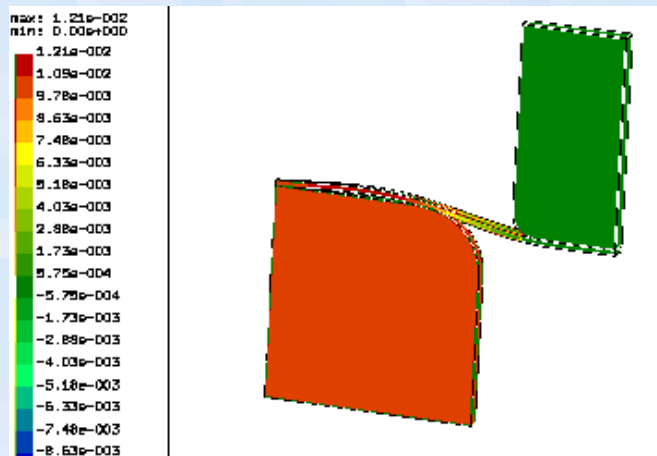
マルチスレッド版で2並列で計算を実行してみる

```
user@user-HP ~/software/ccx213MT
```

```
$ export OMP_NUM_THREADS=2
```

```
user@user-HP ~/software/ccx213MT
```

```
$ time ./CalculiX/ccx_2.13/src/ccx_2.13_MT -i metalformingmortar
```



おおよそ14秒で計算が完了し、
シングルで実行したときより時間がかかってしまった。

```
iterat
```

```
convergence
```

```
the increment size exceeds the remainder of the step and is  
decreased to 0.000000e+00
```

```
pressure ratio (smallest/largest pressure over all contact  
areas) =0.000000e+00
```

```
Using up to 2 cpu(s) for the stress calculation.
```

```
Job finished
```

```
real 0m13.900s
```

```
user 0m13.431s
```

```
sys 0m2.651s
```

Cygwin上での計算実施例

Sample例題実行 MetalForming例題

- 反復法ソルバで計算させてみる。
Input file中のstatic optionでsolverを指定する。
*step,nlgeom,inc=99999
*static, SOLVER=ITERATIVE CHOLESKY

reducing the constant stiffnesses by a factor of
100

*ERROR: too many cutbacks
best solution and residuals are in the frd file

real 2m50.524s
user 3m5.999s
sys 0m40.294s

2分50秒ほど計算して、発散して終了する。
→
やっぱり接触問題は反復法では計算できないみたい？

各行列ソルバの計算速度比較

- 今回は下記のソルバだけの比較ができた。(TIME コマンドで取得)。

ソルバ	計算手法	計算時間	備考
SPOOLES 1CPU	直接法	9.218s	CALCULIX標準
SPOOLES 2CPU	直接法	13.9s	CALCULIX標準
CG法不完全LU	反復法	2m50.524s	CALCULIX標準
TAUCS	直接法	非対称問題なので計算できない とのこと	CALCULIX標準 (OPTION)

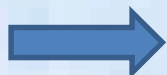
残念ながら、標準直接法ソルバSPOOLES をシングルで使うのが最も早いという結果になった

全体報告まとめ

- **CALCULIX のWINDOWS上のCYGWINを使ったビルド紹介を行った。**
- **CYGWIN上でも通常のLINUXと同様に問題なくビルドできることを確認**
- **今後の予定**
 - **WINDOWS:MINGW環境でのビルド**
 - **IntelMKL のParadiso ソルバの組み込み**
 - **EXTRAS版のビルド**
 - **ソルバTAUCSの組み込み**
 - **ABAQUS / CCXファイルコンバータ開発**

おまけ)MinGWでのエラー

```
$ make
cc -Wall -O3 -fopenmp -I ../.././spooles -DARCH="Linux" -DSPOOLES -DARPACK -DMATRIXSTORAGE -c ccx_2.13.c
ccx_2.13.c:19:3: warning: data definition has no type or storage class
  _set_output_format(_TWO_DIGIT_EXPONENT);
  ^
ccx_2.13.c:19:3: warning: type defaults to 'int' in declaration of '_set_output_format' [-Wimplicit-int]
ccx_2.13.c:19:3: warning: parameter names (without types) in function declaration
In file included from ccx_2.13.c:28:0:
C:/Program Files/blueCFD-Core-2016/msys64/mingw64/x86_64-w64-mingw32/include/stdio.h:519:32: error:
conflicting types for '_set_output_format'
  _CRTIMP unsigned int __cdecl _set_output_format(unsigned int _Format);
  ^
ccx_2.13.c:19:3: note: previous declaration of '_set_output_format' was here
  _set_output_format(_TWO_DIGIT_EXPONENT);
  ^
ccx_2.13.c: In function 'main':
ccx_2.13.c:123:1: warning: implicit declaration of function 'setenv' [-Wimplicit-function-declaration]
setenv("CCX_JOBNAME_GETJOBNAME",jobnamec,1);
^
Makefile:9: ターゲット 'ccx_2.13.o' のレシピで失敗しました
make: *** [ccx_2.13.o] エラー 1
```



ソースのPATCH修成が必要: 気が向いたら修正内容を報告