

OpenFOAM on Eclipse

- 目的
- Eclipse とは
- EclipseのインストールとCDT導入
- カスタムソルバーをコンパイルする方法
- ケースファイルの実行方法
- ケースファイルのデバッグ方法

【動作環境】

DEXCS2011 for OpenFOAM(R)

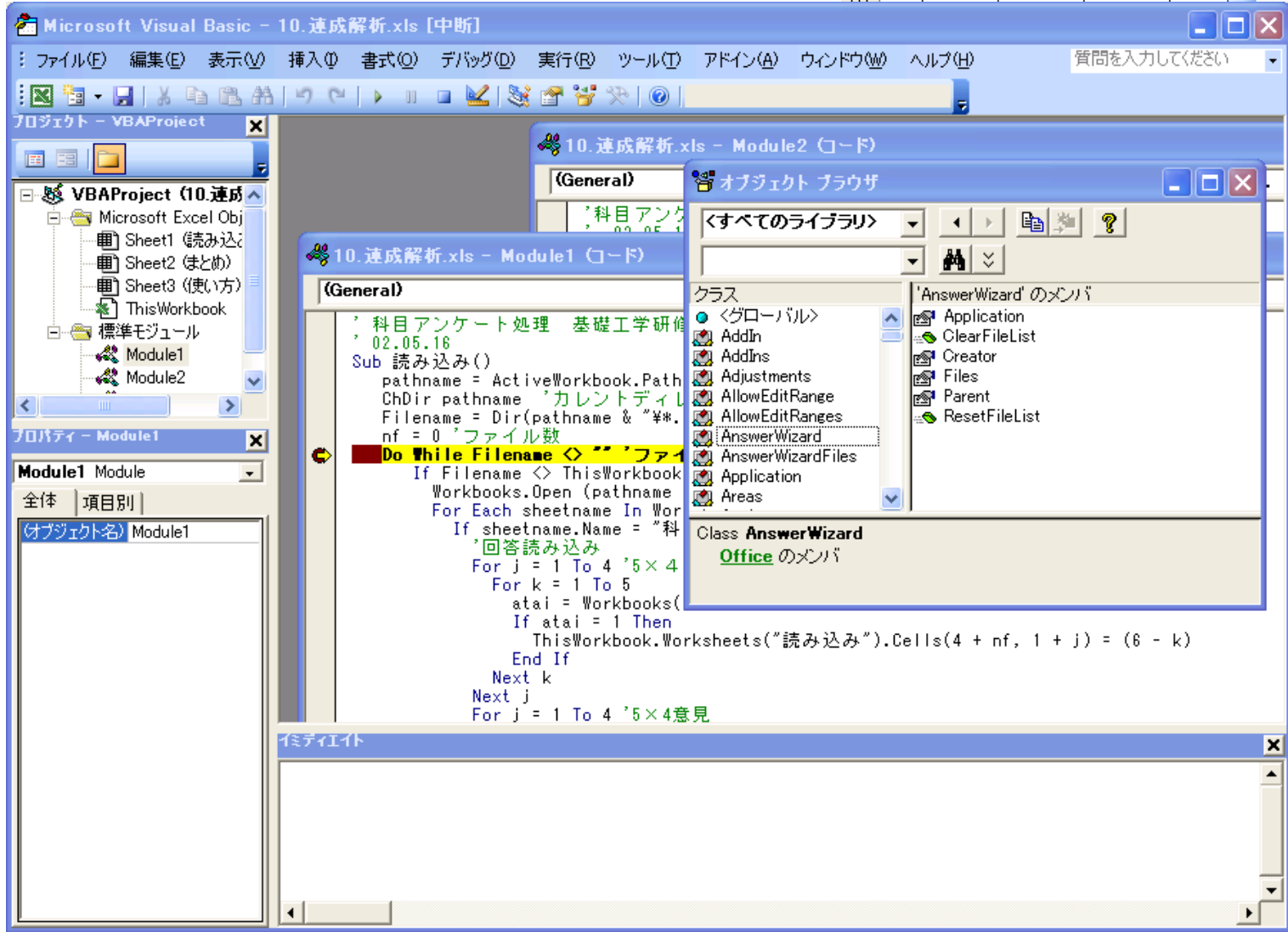
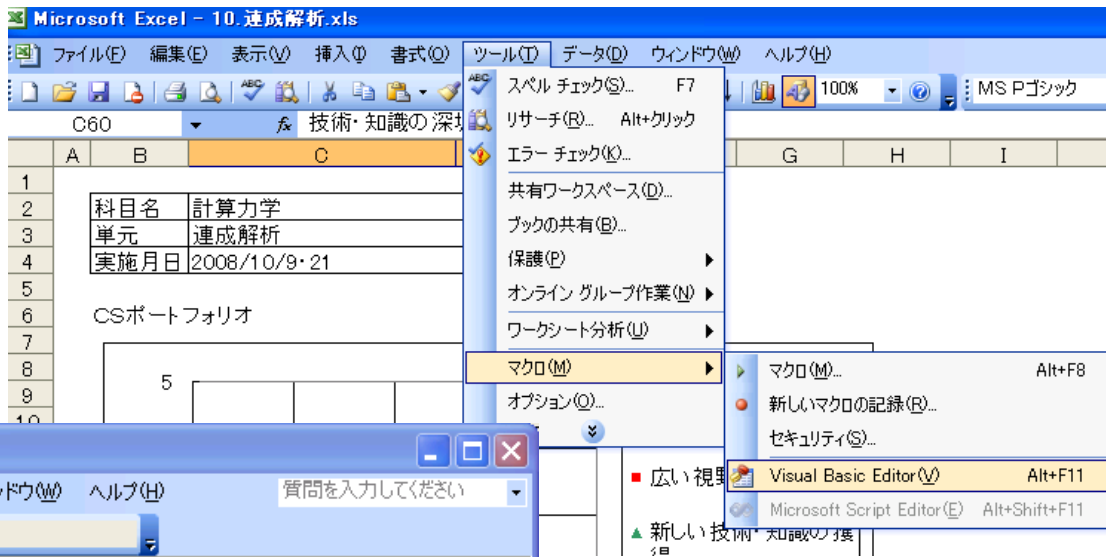
Ubuntu-10.04

OpenFOAM-2.0.x, 2.1.x

目的

- OpenFOAMのカスタムソルバー開発の効率化
 - 統合開発環境の利用、習得

統合開発環境の一例



GUI
エクスプローラー
階層化表示
予約語、関数名補完
エラー表示、参照
デバッグ、ステップ実行

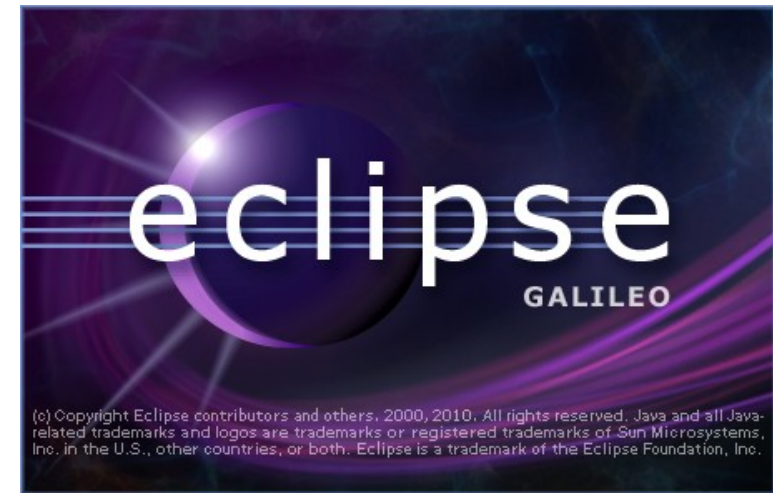
Eclipseとは

参考資料1

What is Eclipse?

Features

- Well-arranged graphical user interface offering project explorer, outline, ...
- Fully integrated powerful text editor offering code highlighting, autocompletion, ...
- Integrated compiler offering linked error and warning marks
- Integrated debugger and debugging environment offering breakpoints and variable information
- Project management: bookmarks and tasks
- Extensions: version management, multiple language support (Java, Python, ...)



Name	
+ <input type="checkbox"/>	Business Intelligence, Reporting and Charting
+ <input type="checkbox"/>	Collaboration
+ <input type="checkbox"/>	Database Development
+ <input type="checkbox"/>	EclipseRT Target Platform Components
+ <input type="checkbox"/>	General Purpose Tools
+ <input type="checkbox"/>	Mobile and Device Development
+ <input type="checkbox"/>	Modeling
- <input type="checkbox"/>	Programming Languages
	<input type="checkbox"/> Dynamic Languages Toolkit - iTcl Development Tools
	<input type="checkbox"/> Dynamic Languages Toolkit - Ruby Development Tools
	<input type="checkbox"/> Dynamic Languages Toolkit - TCL Development Tools
	<input type="checkbox"/> Dynamic Languages Toolkit - XOTcl Development Tools
	<input type="checkbox"/> Eclipse C/C++ Development Tools
	<input type="checkbox"/> Eclipse Java Development Tools
	<input type="checkbox"/> Eclipse XML Editors and Tools
	<input type="checkbox"/> JavaScript Developer Tools
	<input type="checkbox"/> PHP Development Tools (PDT) SDK Feature
+ <input type="checkbox"/>	SOA Development
+ <input type="checkbox"/>	Test and Performance
+ <input type="checkbox"/>	Web, XML, and Java EE Development

Eclipse本

本 > "Eclipse"

関連サーチ: [android](#), [eclipse java](#), [java](#).

検索結果710件中1件から12件までを表示

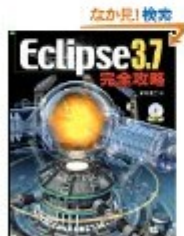
フォーマット

単行本
(187)

単行本(ソフトカバー)
(24)

文庫
(4)

1.



Eclipse 3.7 完全攻略 宮本 信二 (大型)

新品: ¥ 2,940

12時間以内に「お急ぎ便」でご注文いただくと、

★★★★☆ (4)

通常配送無料

目次引用: "... Chapte

本 > "eclipse c++"

5件の検索結果を表示

1.



EclipseではじめるC++ NRIラーニ

新品: ¥ 2,289

10 中古品 ¥ 757より

12時間以内に「お急ぎ便」でご注文いただ

1点在庫あり。ご注文はお早めに。

★★★★☆ (3)

通常配送無料

EclipseではじめるC++



翔泳社
et (2007/06/10)

仕事でC++を使わざるを得ない状況になってきたのと、この際、どうせならEclipseなる開発環境もマスターしたいと思って、一挙両得のタイトルに惹かれて購入。各章とも全て共通フレームで記述されており、コンテンツが少しずつ変わっているだけ。読みやすいと見るべきか、手抜き本といったほうが良いのか。。。ちよっと簡単すぎて、これだけかよ。。。という感がなきにしもあらずでしたが、クラス、コンストラクタ、オーバーロード、継承といったあたりの使い方がようやく実物を通して理解できました。

ちなみに、C++の文法だとか、Eclipseの使い方だとかは、ほとんど説明ありません。まずは慣れる！使えそうに思ったら改めて別の本買って勉強せい！ということのようです。

(2007/6/10)

<http://mogura7.zenno.info/~et/xoops/modules/amaxoop2/article.php?lid=5080>



在庫あり。

¥2,289 | 発売:2007-01-25 | セールスランク:715133

コメント(0) | クリック(444) | 評価(6.00 / 1) -- 10点満点

ソースコードの解読ツールとしての利用

<http://www.geocities.co.jp/SiliconValley-SantaClara/1183/study/OpenFOAM/eclipse.html>

Eclipse による OpenFOAM ソースコードの探索

2010年11月3日

使い方

1. たとえば、画面左の Project Explorer から simpleFoam のソースコード (OpenFOAM/OpenFOAM-1.7.1/applications/solvers/incompressible/simpleFoam/simpleFoam.C) を開く。
2. Ctrl を押しながら

```
#include "createFields.H"
```

を左クリックすると、createFields.H が表示される。

3. createFields.H の

```
volScalarField p
```

の "volScalarField" を Ctrl + 左クリックすると、volFieldsFwd.H が開き volScalarField の定義が表示される。

- Ctrl + 左クリックの代わりに F3 でもよい。たまに複数の候補が提示される場合があるが、それっぽいのを 1 つ選べばよい。表示する候補を 1 つも見つけれない場合もある。
- 画面右の Outline を使うと、ファイル内の関数などへのアクセスが速くて便利。
- クラス名にカーソルを合わせて F4 を押すと、クラス階層 (Type Hierarchy) が表示される。表示できない場合もあるが、親クラスや子クラスでクラス階層を表示してやるとうまくいったりする。
- 表示しているファイルを画面左の Project Explorer に表示するには、右クリックのポップアップメニューで [Show In]-[Project Explorer] を選ぶ。

探索例

simpleFoam の運動方程式の左辺はつぎのように表されている (UEqn.H)

```
tmp<fvVectorMatrix> UEqn  
(  
    fvm::div(phi, U)  
    + turbulence->divDevReff(U)
```

6th OpenFOAM Workshop

13-16 June 2011



6th OpenFOAM Workshop 13-16 June 2011

PROGRAM

Day	Date	Time	Event
<u>Monday</u>	June 13	7:30-8:30	Registration and Check-In
		8:00-8:30	Use of USB Stick
		8:30-4:30	<u>OpenFOAM Workshop Training Sessions</u>
<u>Tuesday</u>	June 14	7:30-8:00	Breakfast
		8:00-10:10	Technical Presentations
		10:10-10:30	Poster Session
		11:00-5:00	Technical Presentations
		All breaks	Posters on display
<u>Wednesday</u>	June 15	7:30-8:00	Breakfast
		8:00-9:00	Feature Talk
		9:00-4:40	Technical Presentations
		All breaks	Posters on display
		6:30-10:00	Banquet
<u>Thursday</u>	June 16	8:30-4:30	Special Interest Group (SIG) and Birds-of-a-Feather (BOF) meetings

6th OpenFOAM Workshop - Training Sessions - Monday, 13 June 2011

New User	New Developer	Visualization
Advanced Applications	Advanced Developer	Pre/Post Processing

Time	Session A (213 IST Cybertorium) - 100 seats	Session B (116 EES) - 30 seats	Session C (119 EES) - 49 seats	Session D (121 EES) - 49 seats
8:30 - 10:00	<p>Block-coupled solvers, I. Clifford, PSU</p> <p>Abstract Slides Files</p>	<p>Using Blender with OpenFOAM to produce high quality renderings, Matt Cragun, Totalsim Inc.</p> <p>Abstract Slides Files</p>	<p>Turbomachinery CFD with OpenFOAM, H. Jasak, Wikki Ltd</p> <p>Abstract Slides Files</p>	<p>"I don't even know where to start with OpenFOAM," D. Boger, PSU</p> <p>Abstract Slides Files</p>
10:30 - noon	<p>Adaptive Tetrahedral Re-Meshing for Deforming Domains, K. Mooney, UMass Amherst</p> <p>Abstract Slides Files</p>	<p>Integrated Development Environment Eclipse (IDE) for OpenFOAM Assessing the Development and Performance of bubbleFoam, Astrid Mahrla and Holger Marschall (TU Munchen and TU Darmstadt)</p> <p>Abstract Slides Files</p>	<p>NREL Wind Farm Tool and Tutorials, M. Churchfield, National Renewable Energy Lab</p> <p>Abstract Slides Files</p>	<p>Precompiled Apps/Utils. Running Tutorials, Running in Parallel, and General Post-Processing Utilities, Eugene de Villiers, engys Ltd.</p> <p>Abstract Slides</p>
1:00 - 2:30	<p>Multi-Zonal Multi-Physics Modeling: FSI and Conjugate Heat/Mass Transfer, R. Campbell and B. Craven, PSU</p> <p>Abstract Slides Files</p>	<p>Ship Resistance and Propulsion Simulations with OpenFOAM, K. Maki, U. Michigan</p> <p>Abstract Slides Files</p>	<p>Multi-block structured meshing and pre-processing for OpenFOAM turbomachinery analysis, C. Sideroff, Pointwise</p> <p>Abstract Slides Files</p>	<p>How to add a transport equation to scalarTransportFoam, H. Rusche, Wikki Ltd</p> <p>Abstract Slides Files</p>
3:00 - 4:30	<p>Five Basic Classes of OpenFOAM, H. Jasak, Wikki Ltd.</p> <p>Abstract Slides</p>	<p>Simulating the Combustion of Gaseous Fuels, Dominik Christ, RWTH Aachen University</p> <p>Abstract Slides Files</p>	<p>"No C++, please. We're users!" Case setup and quantitative evaluation using swak4Foam (and a bit of PyFoam), Bernhard Gschaider, ICE Strömungsforschung GmbH</p> <p>Abstract Slides Files</p>	<p>Advanced Paraview Tutorial, Dave DeMarle, Kitware, Inc.</p> <p>Abstract Slides Files</p>

参考資料1

OpenFOAM Workshop Training Session

Integrated Development Environment (IDE) Eclipse for OpenFOAM®

Astrid Mahrla¹ and Holger Marschall²

¹Chair of Chemical Engineering, Technische Universität München
²Center of Smart Interfaces, Technische Universität Darmstadt
Germany

6th OpenFOAM Workshop
13 - 16 June 2011
The Pennsylvania State University
State College, PA, USA

しかし・・・挫折

参考資料2



Integrated Development Environment (IDE) Eclipse for OpenFOAM®

Assessing the Performance of bubbleFoam

Astrid Mahrla¹ and Holger Marschall²

¹Chair of Chemical Engineering, Technische Universität München
²Center of Smart Interfaces, Technische Universität Darmstadt

June 13, 2011

1 Introduction

"Eclipse is an open source community whose projects are focused on building an extensible development platform, runtimes and application frameworks for building, deploying and managing software across the entire software lifecycle. Many people know us, and hopefully love us, as a Java IDE but Eclipse is much more than a Java IDE." - www.eclipse.org

Scope and objective of this tutorial is the introduction of the *Integrated Development Environment (IDE) Eclipse* for OpenFOAM®. Eclipse is a powerful IDE originally developed for Java programming. But with the *C/C++ Development ToolKit (CDT)* extension Eclipse becomes a very common IDE for *fast and efficient C++* programming.

Due to the amount of advantages of using an IDE only a few of them are listed below.

- Well-arranged graphical user interface offering project explorer, outline, ...
- Fully integrated powerful text editor offering code highlighting, autocompletion, ...
- Integrated compiler offering linked error and warning marks
- Integrated debugger and debugging environment offering breakpoints and variable information
- Project management: bookmarks and tasks
- Extension: version management, multiple language support (Java, Python, ...)

¹Corresponding Author: Astrid Mahrla (Astrid.Mahrla@ch.tum.de)







Page [Discussion](#) [View](#) [View source](#) [History](#)

Go

Search

Howto Use OpenFOAM with Eclipse

Using Eclipse CDT for developing OpenFOAM

Please note: This offering is not approved or endorsed by [OpenCFD®](#)  Limited, the producer of [OpenFOAM®](#)  software and owner of the [OpenFOAM®](#)  and [OpenCFD®](#)  trademarks.

Contents

- 1 Thanks to ..
- 2 Versions
- 3 Eclipse Capabilities
- 4 Aim of the tutorial
- 5 Download and set up Eclipse
 - 5.1 Install CDT by hand
 - 5.2 Launching C++ project
- 6 Developing and compiling
- 7 Running applications and debugging
 - 7.1 Debugging in Parallel
- 8 Profiling with gProf and Linux Tools
- 9 Annotations and hints
 - 9.1 Eclipse and python
 - 9.2 Working in parallel with the terminal
 - 9.3 Increase Java heap space for Eclipse

再度挑戰!

Navigation

- [Main page](#)
- [Community portal](#)
- [Current events](#)
- [Recent changes](#)
- [Random page](#)
- [FAQ](#)
- [Article Categories](#)
- [Help](#)

Toolbox

- [What links here](#)
- [Related changes](#)
- [Special pages](#)
- [Permanent link](#)
- [Browse properties](#)

[Print/export](#)

Eclipse上でOpenFOAMを使う・・・とは

- ソースコードの参照(勉強)
- カスタムソルバーのコンパイル
- 上記カスタムソルバーでケースファイルを実行
- 同上、デバッグ

EclipseのインストールとCDT導入

<http://mogura7.zenno.info/~et/xoops/modules/wordpress/index.php?p=444>

2012年1月26日(木曜日)

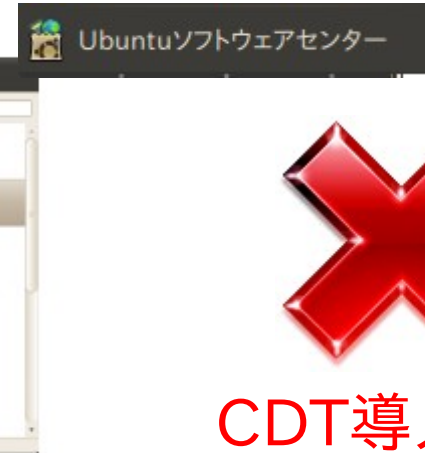
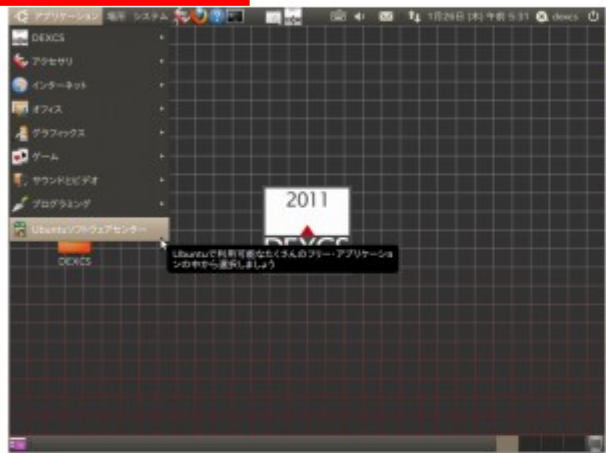
DEXCS2011 for OpenFOAM® にEclipseをインストールする方法

カテゴリー: General CAE OpenFOAM DEXCS - et @ 21時58分48秒

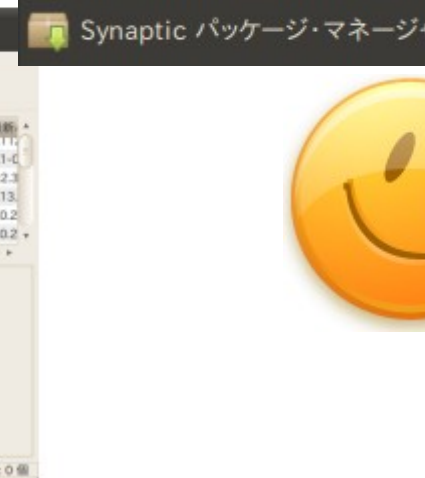
- 昨日の記事でEclipseの使用方法メモがあり、順序が逆になってしまいましたが、Eclipseをインストールする方法です。

Eclipse のインストール

- DEXCS2011 for OpenFOAM(R)のOSは、ubuntu-10.04なので、Eclipseはubuntuソフトウェアセンター、またはSynaptic/パッケージ・マネージャを使えば簡単にインストールが出来る。
- ubuntuソフトウェアセンター



- しかし、この方法だと、後のCDT(C/C++ Development Toolkit)プラグインの導入が出来ない。
- Synapticパッケージ・マネージャによるインストール



CDTの導入

CDT: C/C++ Development Toolkit

The screenshot shows the Eclipse IDE interface. The 'Help' menu is open, and the 'Install' dialog is displayed. Red boxes with numbers 1 through 6 highlight key steps in the process:

- 1: The 'Help' menu is open.
- 2: The 'Install New Software...' option is selected in the 'Help' menu.
- 3: The 'Add...' button is clicked in the 'Install' dialog.
- 4: The 'Work with:' dropdown menu is set to 'Galileo Update Site - http://download.eclipse.org/releases/galileo/'.
- 5: The 'Eclipse C/C++ Development' item is selected in the list of available software.
- 6: The 'Next >' button is clicked in the 'Install' dialog.

The 'Install' dialog shows the following details:

Available Software
Check the items that you wish to install.

Work with: Galileo Update Site - <http://download.eclipse.org/releases/galileo/>

Find more software by working with the '[Available Software Sites](#)' preference

type filter text

Name	Version
<input type="checkbox"/> Dynamic Languages Toolk...	1.0.0.v20090610-1638-7D-4E...
<input type="checkbox"/> Dynamic Languages Toolk...	1.0.1.v20090903-2153-7D-4E...
<input type="checkbox"/> Dynamic Languages Toolk...	1.0.0.v20090610-1638-4-0_...
<input checked="" type="checkbox"/> Eclipse C/C++ Developme...	6.0.2.201002161416
<input type="checkbox"/> Eclipse Java Development...	3.5.2.r352_v20100108-7r88...
<input type="checkbox"/> Eclipse XML Editors and...	3.1.1.v200907161031-7H65...

Details
Eclipse C/C++ development tools. (Binary runtime and user documentation.)

Show only the latest versions of available software Hide items that are already installed

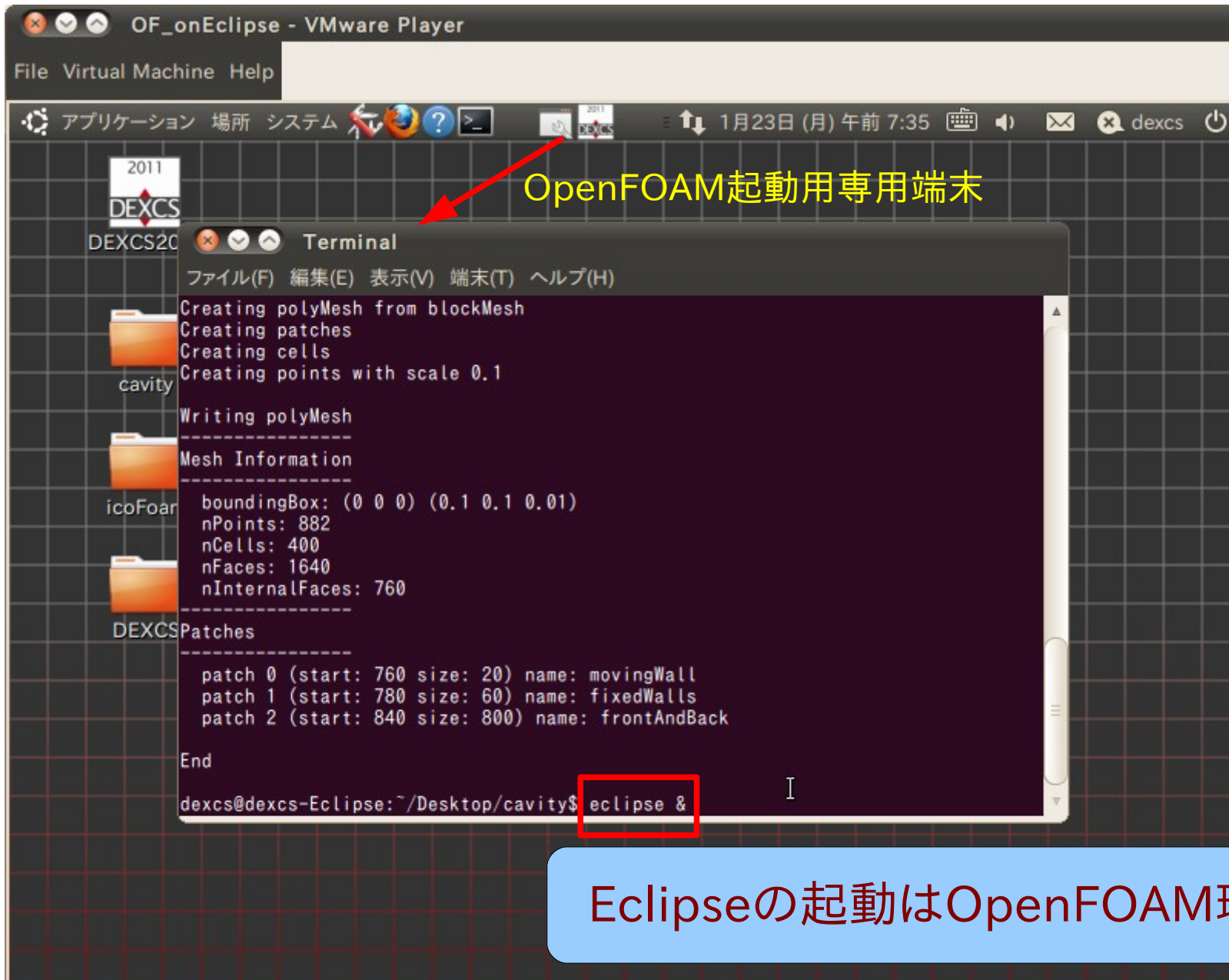
Group items by category [What is already installed?](#)

Contact all update sites during install to find required software

Buttons: ? < Back Next > Cancel Finish

Eclipseの起動方法

<http://mogura7.zenno.info/~et/xoops/modules/wordpress/index.php?p=443>

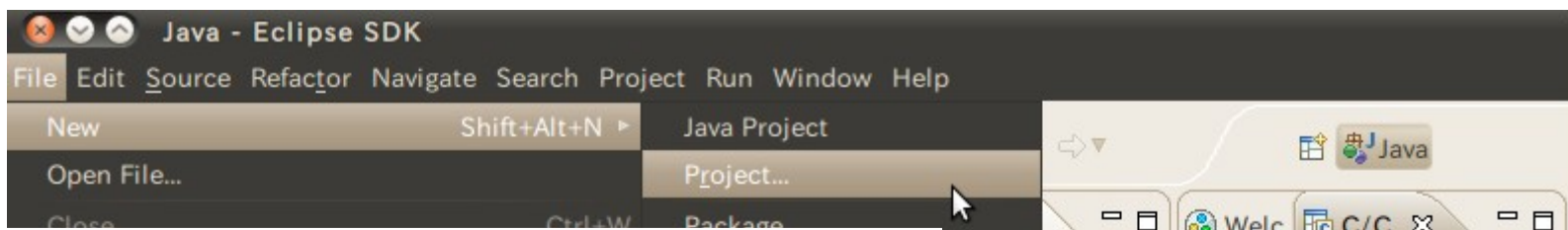


コンパイル方法指定の要点1

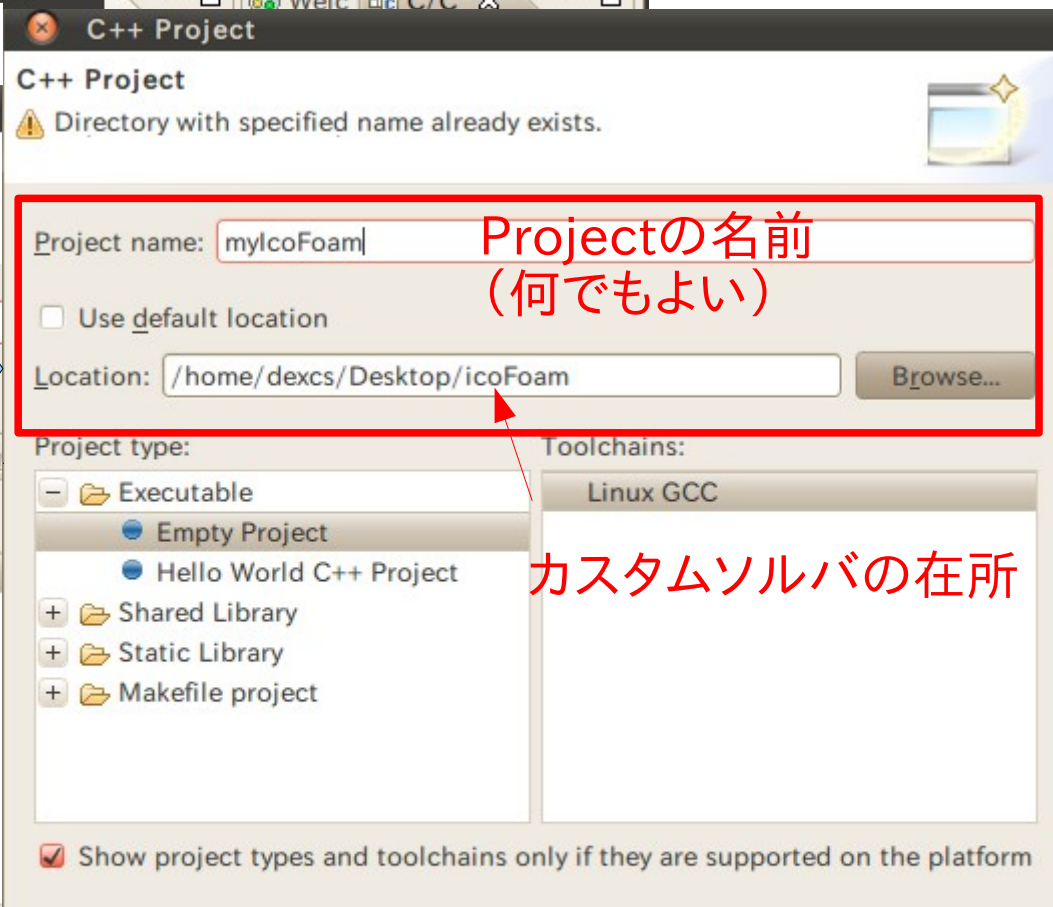
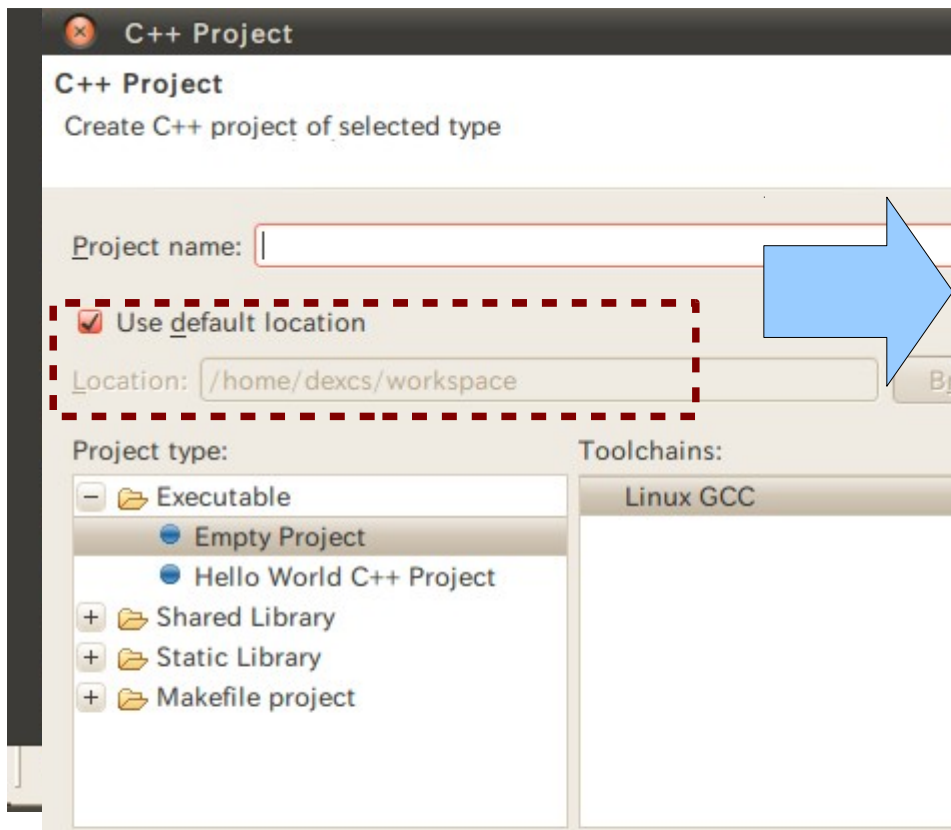


C++通常のBuild(コンパイル)ではない

新規Project (カスタムソルバー作成用)

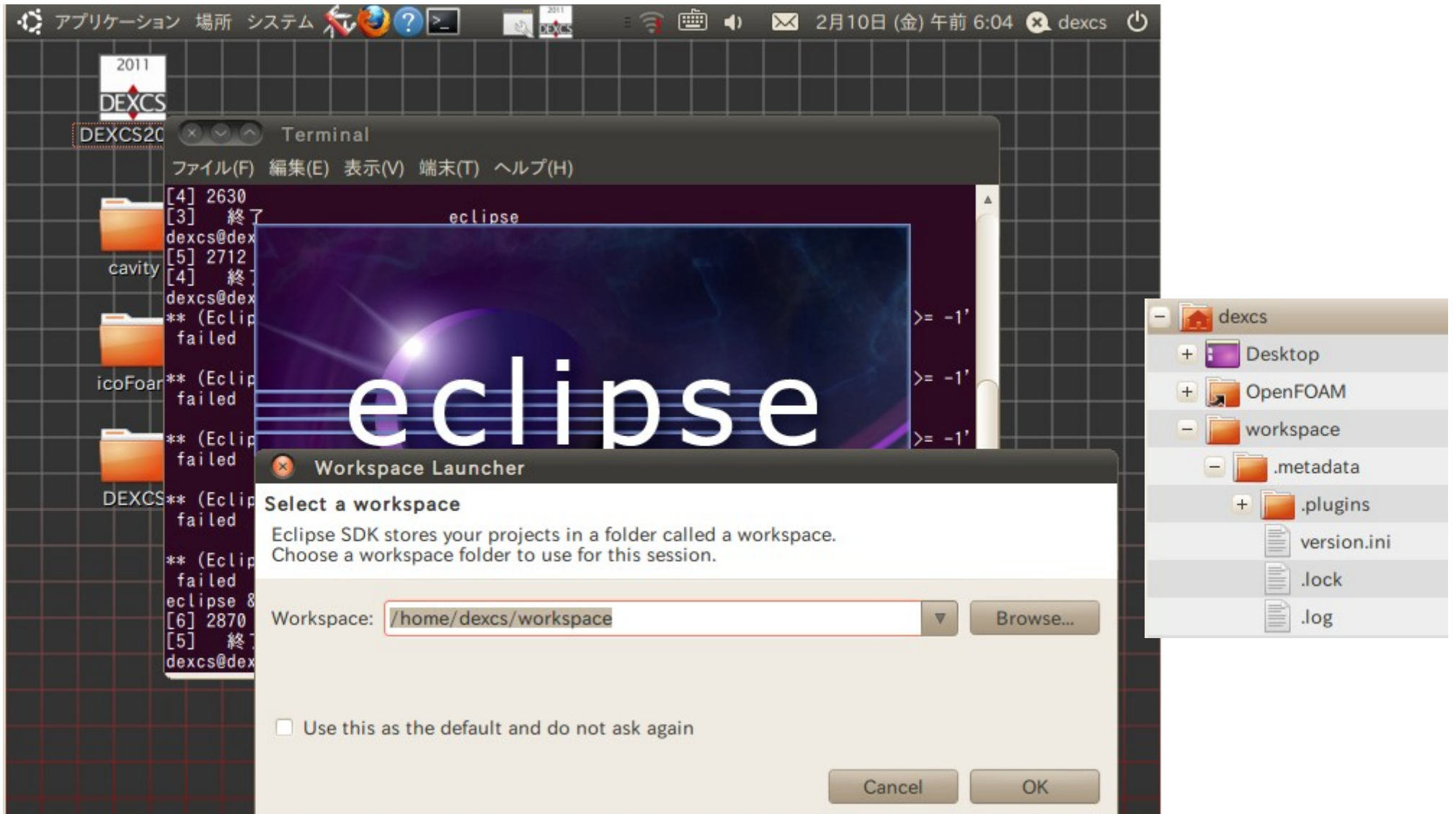


Files→New→Project→C++Project



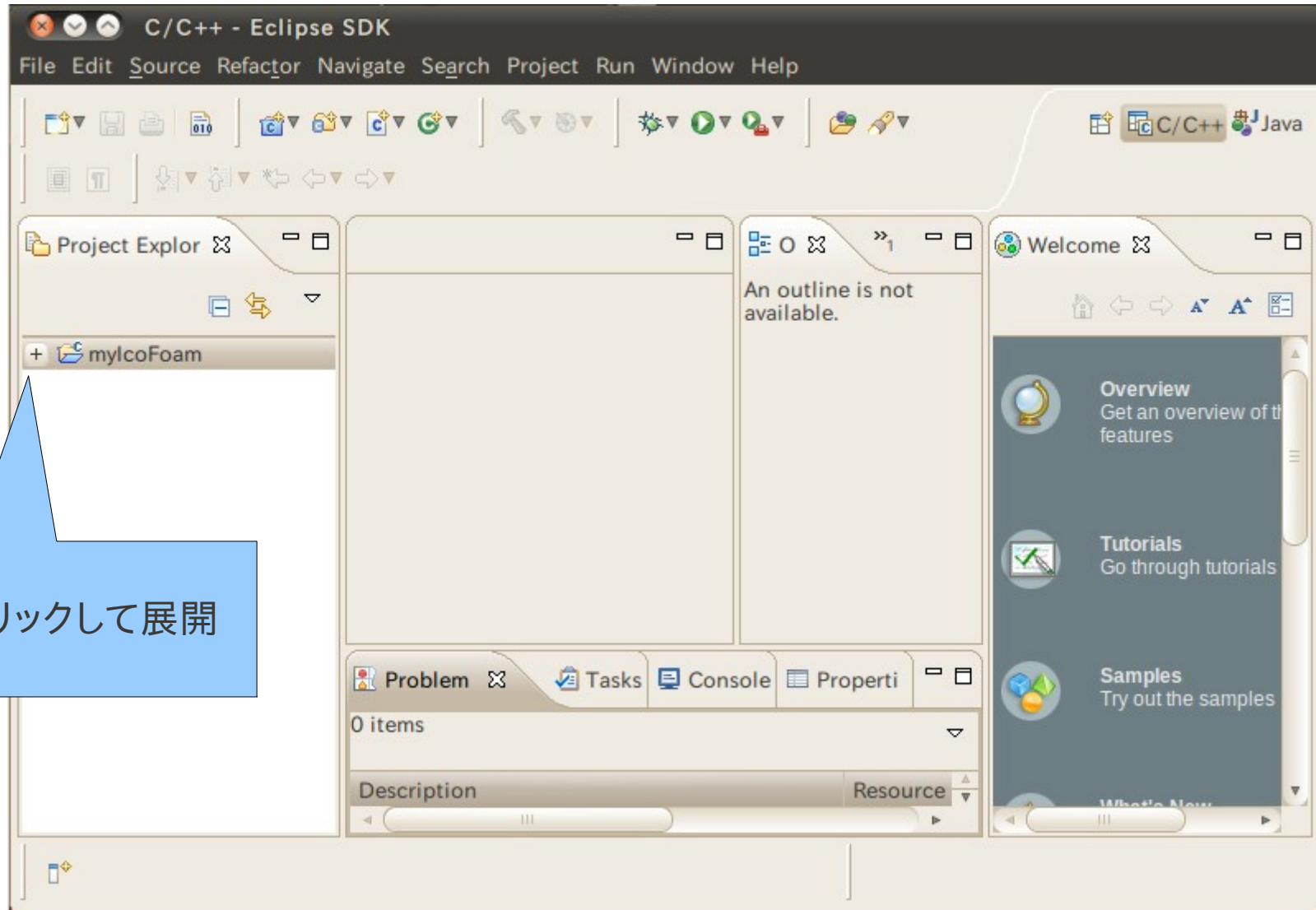
カスタムソルバー毎に Location を変更

(参考) Eclipse起動時Workspace



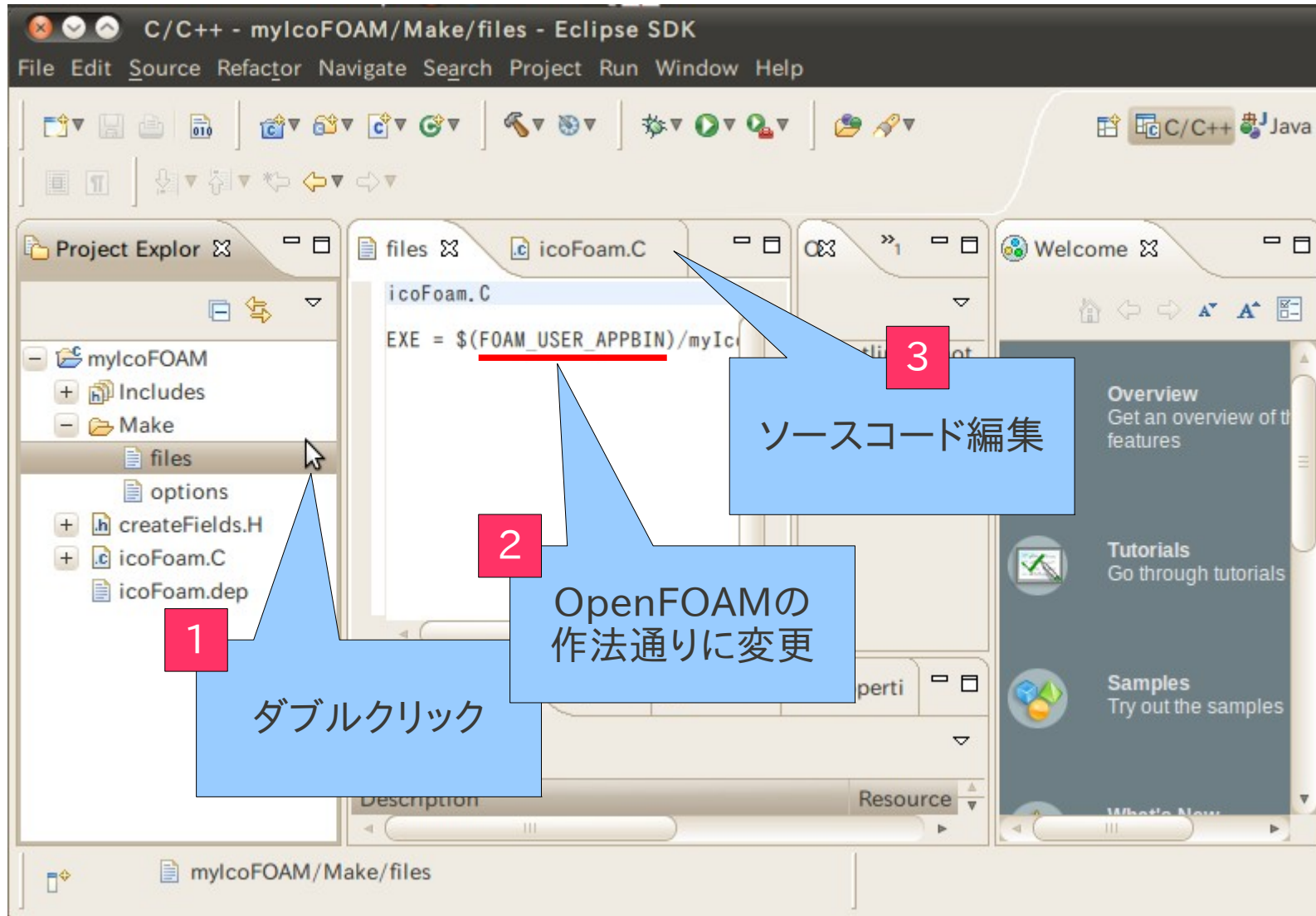
起動時に設定するWorkspaceはEclipse本体の設定項目など

カスタムソルバーの編集

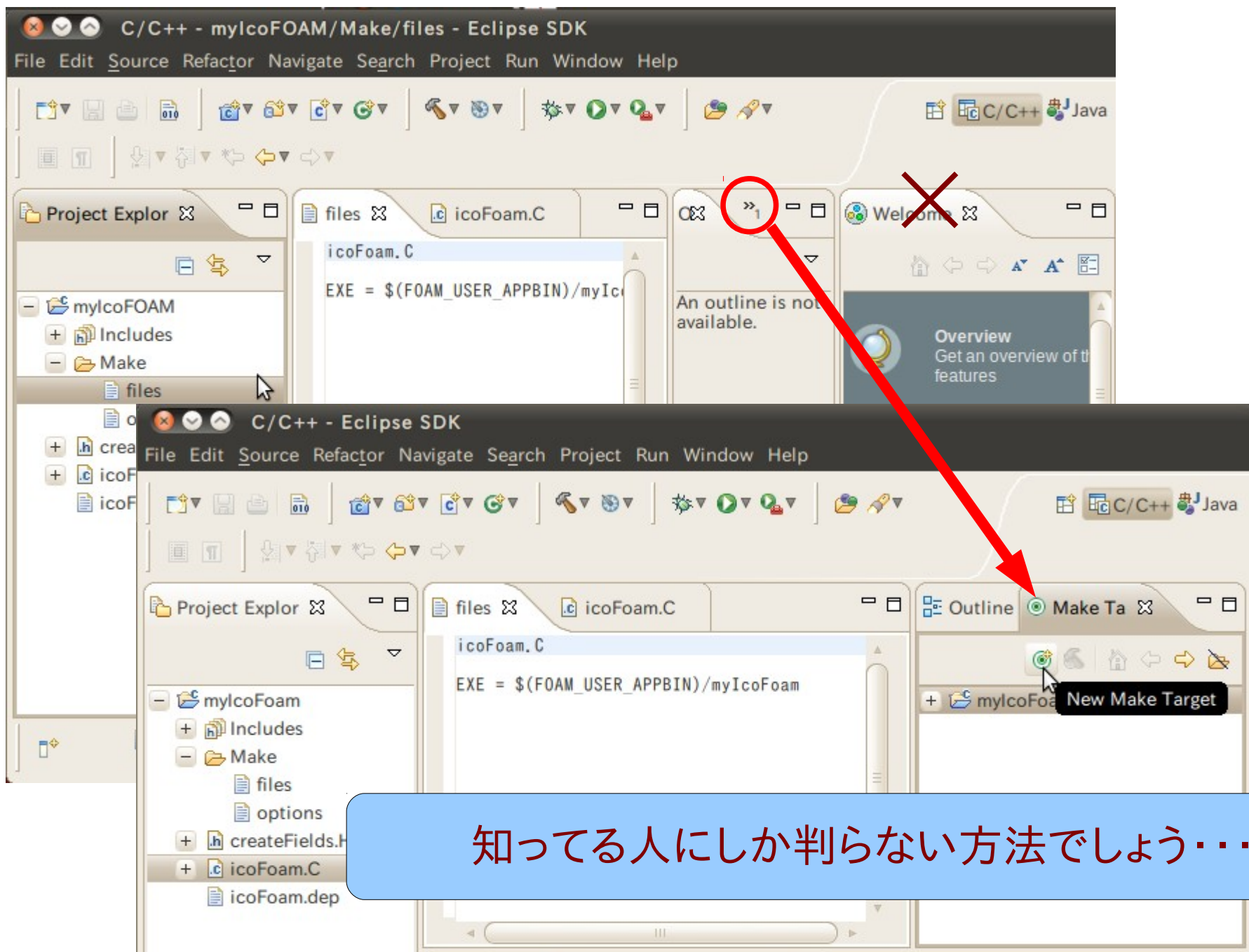


クリックして展開

カスタムソルバーの編集



Build (コンパイル) 方法の指定法



知ってる人にしか判らない方法でしょう...

コンパイル用GUIボタンの作成

C/C++ - Eclipse SDK

File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer: myIcoFoam, Includes, Make, files, options, createFields.H, icoFoam.C, icoFoam.dep

icoFoam.C
EXE = \$(FOAM_USER_APPBIN)/myIcoFoam

1 クリック New Make Target

2 名前 (何でもよい) ブランク wmake

3

Create Make Target

Target name: []

Make Target

Same as the target name

Make target: []

Build Command

Use builder settings

Build command: make

Build Settings

Stop on first build error

Run all project builders

Cancel OK

Create Make Target

Target name: wmake

Make Target

Same as the target name

Make target: []

Build Command

Use builder settings

Build command: wmake

Build Settings

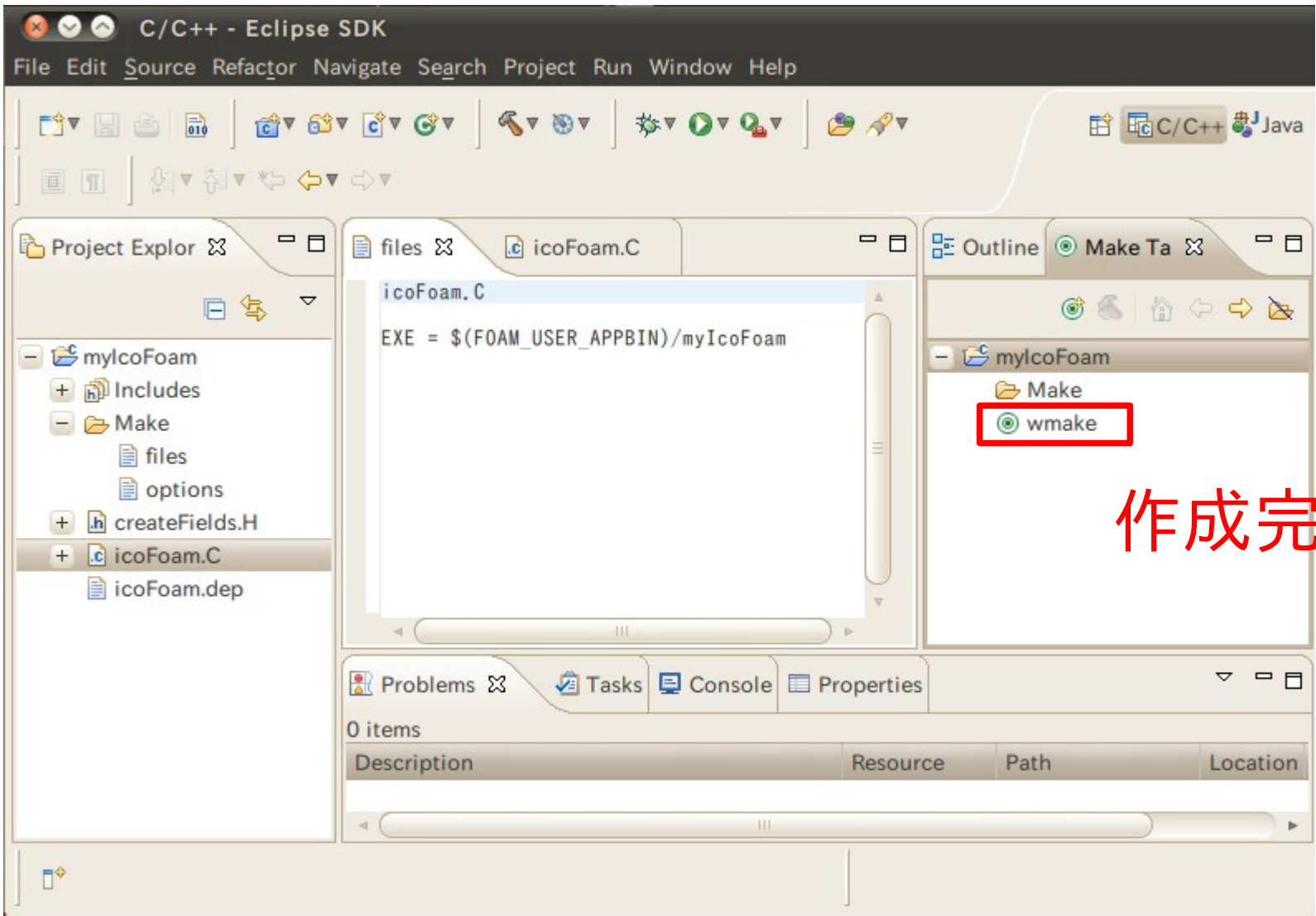
Stop on first build error

Run all project builders

Cancel OK

Must specify a non-empty target name

コンパイル用GUIボタン



コンパイルの実行

1 ダブルクリック

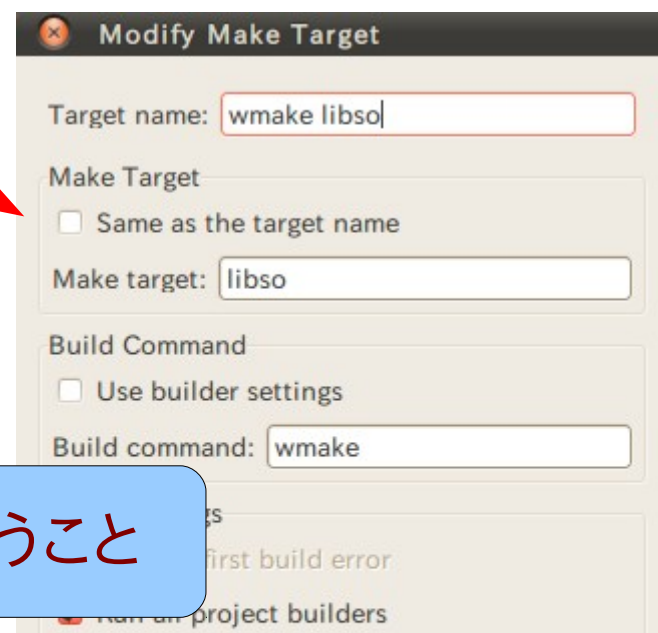
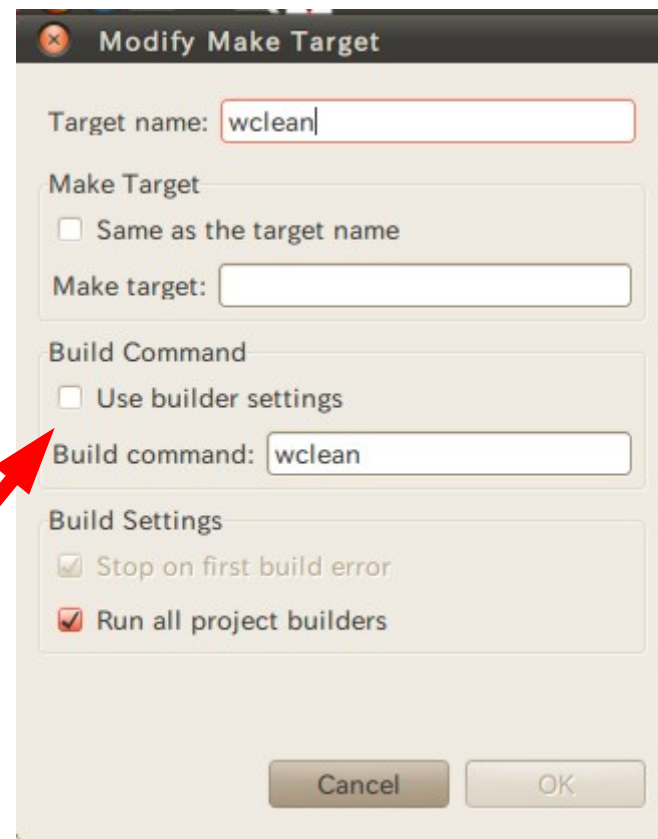
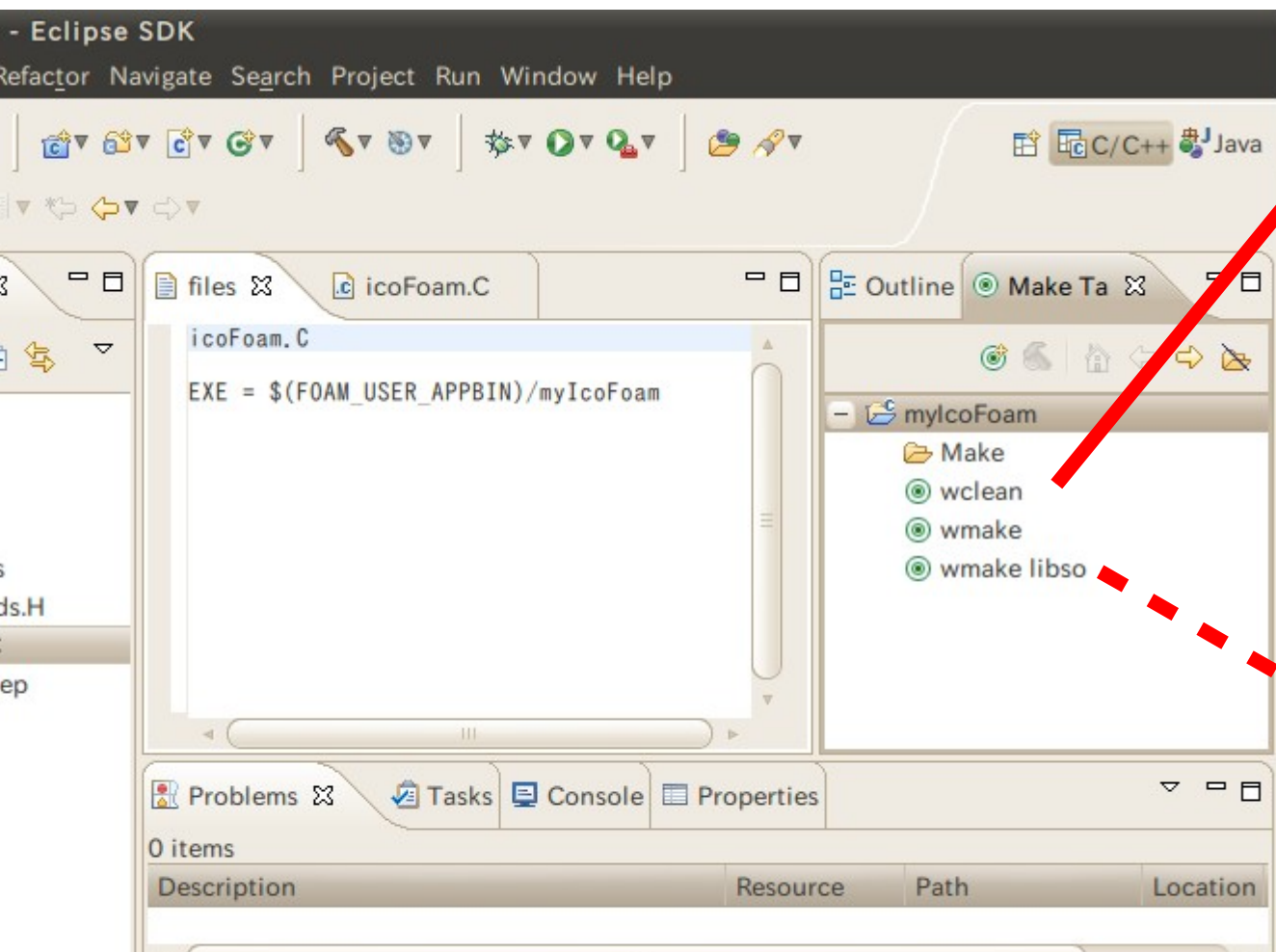
2 コンソール画面

3 コンパイル成功!

```
1 icoFoam.C
2
3 EXE = $(FOAM_USER_APPBIN)/myIcoFoam
4
```

```
wmake
g++ -m64 -Dlinux64 -DWM_DP -Wall -Wextra -Wno-unused-parameter -Wold-style-cast
-Wnon-virtual-dtor -O3 -DNoRepository -ftemplate-depth-100 -I/home/dexcs/
OpenFOAM/OpenFOAM-2.0.x/src/finiteVolume/lnInclude -IlnInclude -I. -I/home/
dexcs/OpenFOAM/OpenFOAM-2.0.x/src/OpenFOAM/lnInclude -I/home/dexcs/OpenFOAM/
OpenFOAM-2.0.x/src/OSspecific/POSIX/lnInclude -fPIC -Xlinker --add-needed
Make/linux64GccDPOpt/icoFoam.o -L/home/dexcs/OpenFOAM/OpenFOAM-2.0.x/platforms/
linux64GccDPOpt/lib ¥
-lfiniteVolume -lOpenFOAM -ldl -lm -o /home/dexcs/OpenFOAM/
dexcs-2.0.x/platforms/linux64GccDPOpt/bin/myIcoFoam
```

wmake, wclean, ...

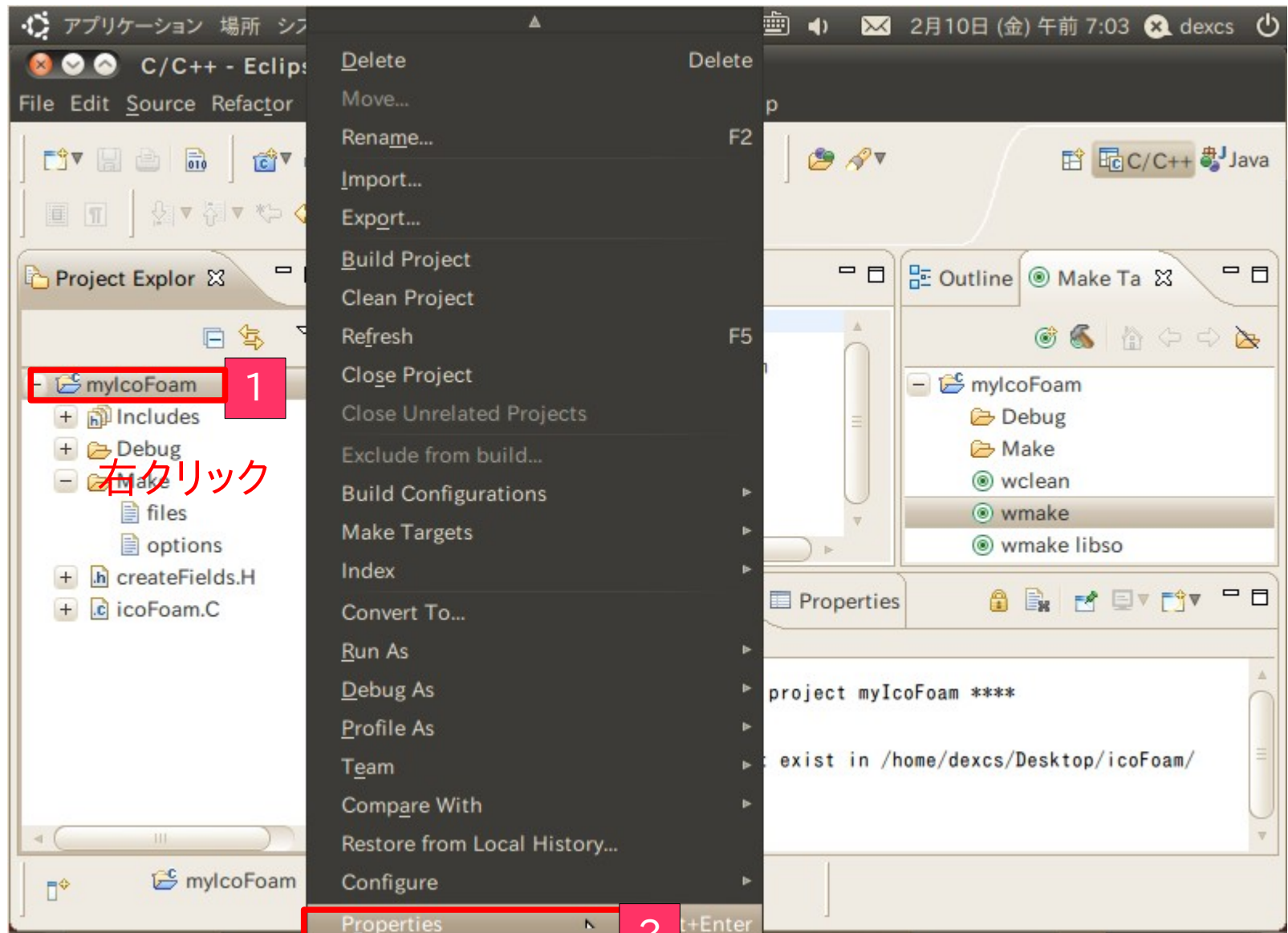


Build はコンパイルするだけじゃないんだということ

よくあるトラブル

The screenshot shows the Eclipse IDE interface for a C/C++ project named 'myIcoFoam'. The Project Explorer on the left shows the project structure with folders for 'Includes', 'Debug', 'Make', 'files', and 'options'. The 'Make' folder is highlighted with a red dashed box. The main editor shows the 'icoFoam.C' file with the line 'EXE = \$(FOAM_USER_APPBIN)/myIcoFoam'. The Console window at the bottom shows the output of a build command, with the error message 'wmake error: 'Make' directory does not exist in /home/dexcs/Desktop/icoFoam/Debug' highlighted in red. A blue question mark icon is positioned in the bottom left corner of the IDE window.

よくあるトラブル⇒対策



Propertiesを選択

Build directory の確認

The screenshot shows the 'Properties for mylcoFoam' dialog box. The 'C/C++ Build' section is active, and the 'Behaviour' tab is selected. The 'Build location' section is highlighted with a red dashed box. A blue arrow points from this section to a red-bordered inset showing the 'Build directory' field set to '/home/dexcs/Desktop/icoFoam'.

Properties for mylcoFoam

type filter text

Resource
Builders
+ C/C++ Build
+ C/C++ General
Project References
Run/Debug Settings

C/C++ Build

Configuration: Debug [Active] Manage Configurations...

Builder Settings Behaviour

Builder
Builder type: External builder
 Use default build command
Build command: make Variables...

Makefile generation
 Generate Makefiles automatically Expand Env. Variable Refs in Makefiles

Build location
Build directory: \${workspace_loc:/mylcoFoam/Debug}

Makefile generation
 Generate Makefiles automatically Expand Env. Variable Refs in Makefiles

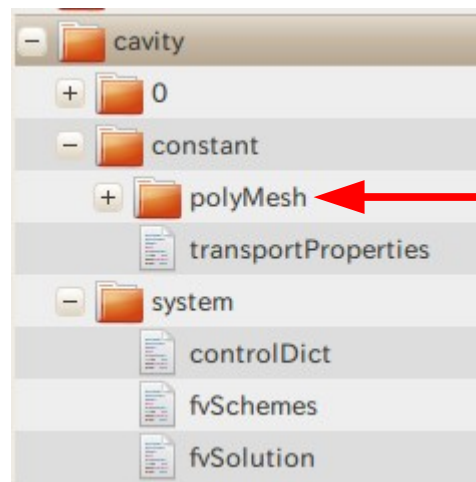
Build location
Build directory: /home/dexcs/Desktop/icoFoam
Workspace... File system... Variables...

Project Location(work directory)
ソースコード、Makeフォルダの在所

Eclipse上でOpenFOAMを使う・・・とは

- ソースコードの参照(勉強)
- カスタムソルバーのコンパイル
- 上記カスタムソルバーでケースファイルを実行
- 同上、デバッグ

ケースファイルの実行準備



メッシュは作成済であること

ケースファイルのImport

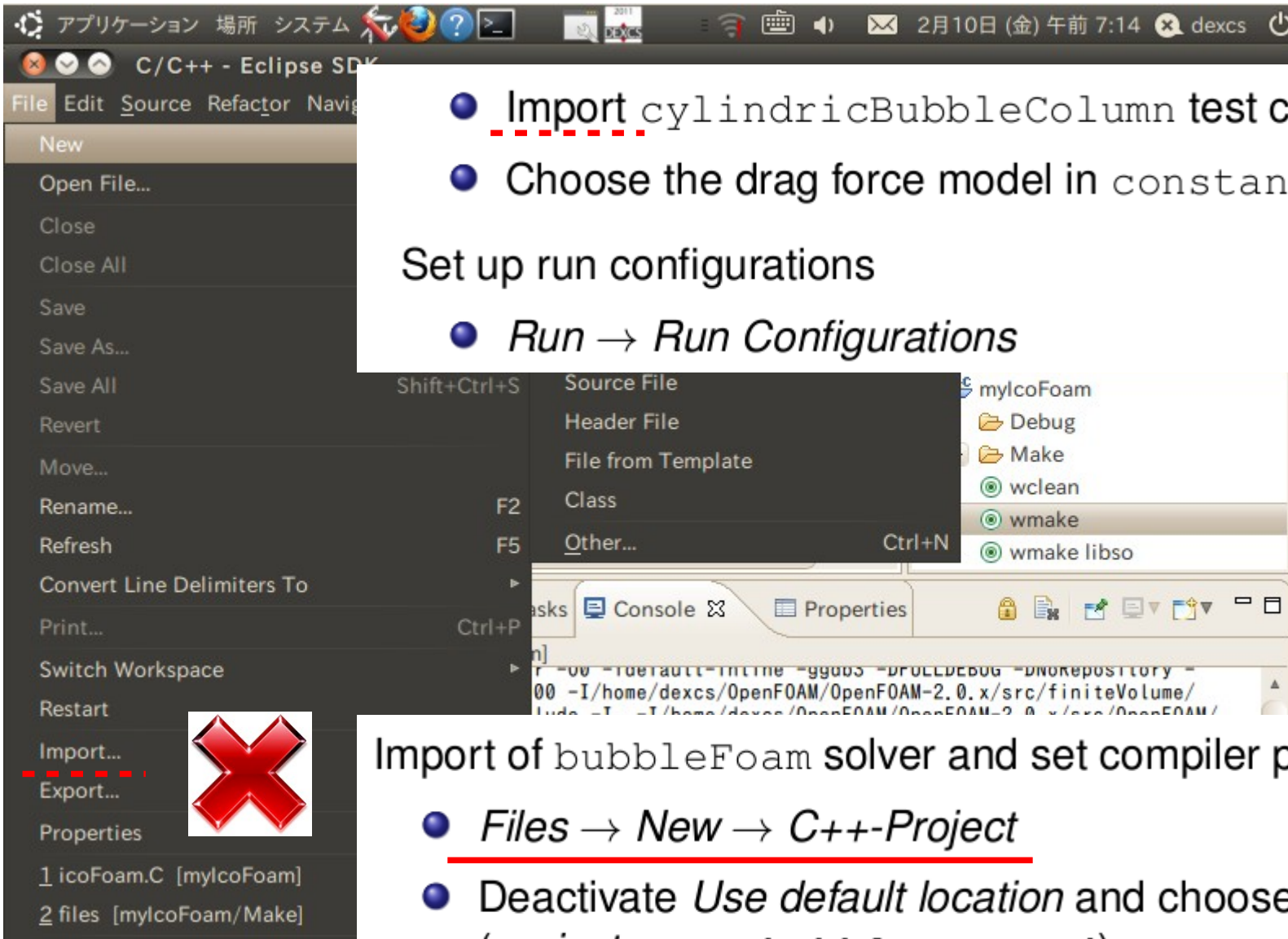
参考資料1

- Import `cylindricBubbleColumn` test case

- Choose the drag force model in `constant/interfacialPro`

Set up run configurations

- *Run* → *Run Configurations*



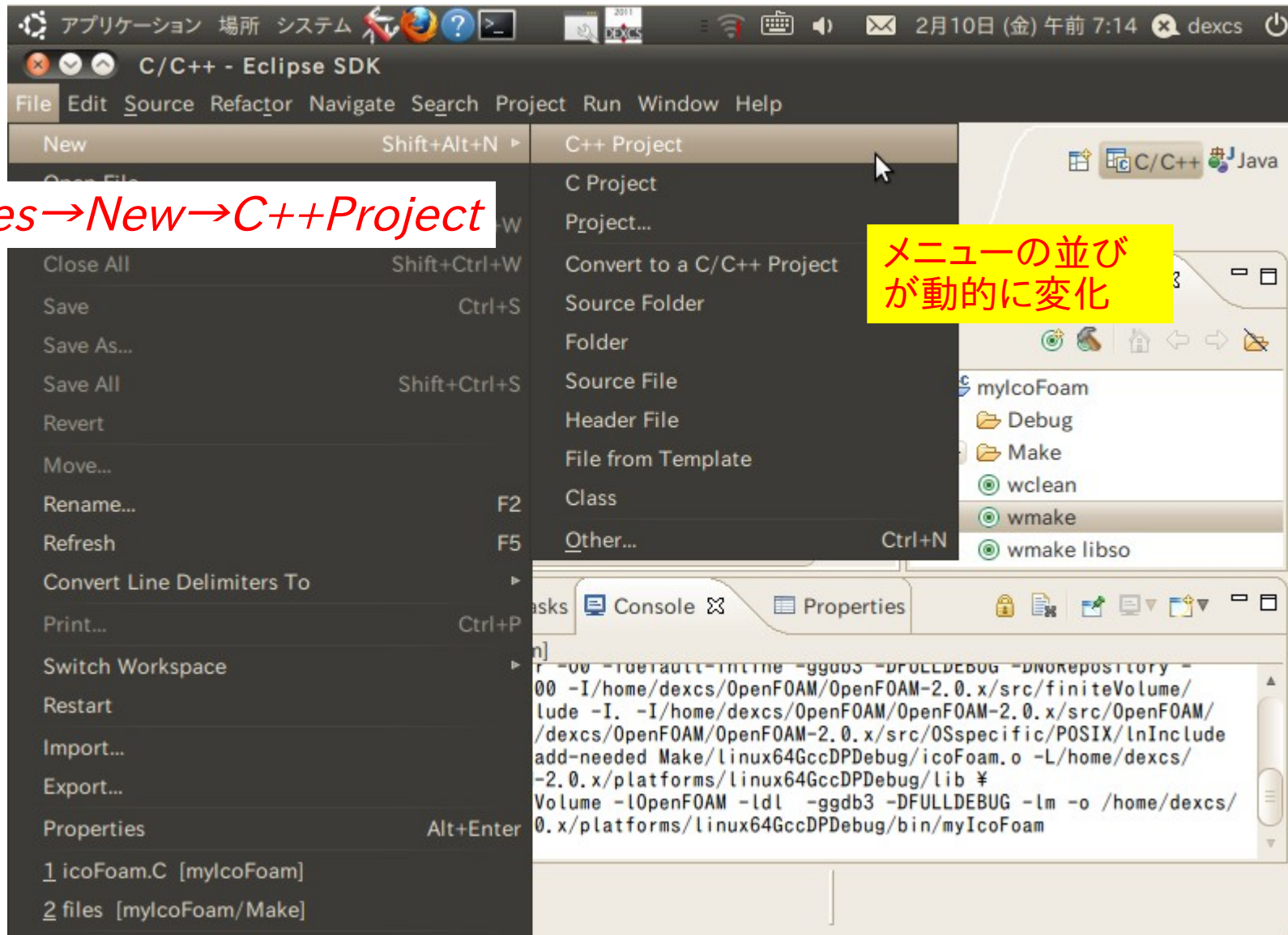
Import of `bubbleFoam` solver and set compiler properties

- *Files* → *New* → *C++-Project*

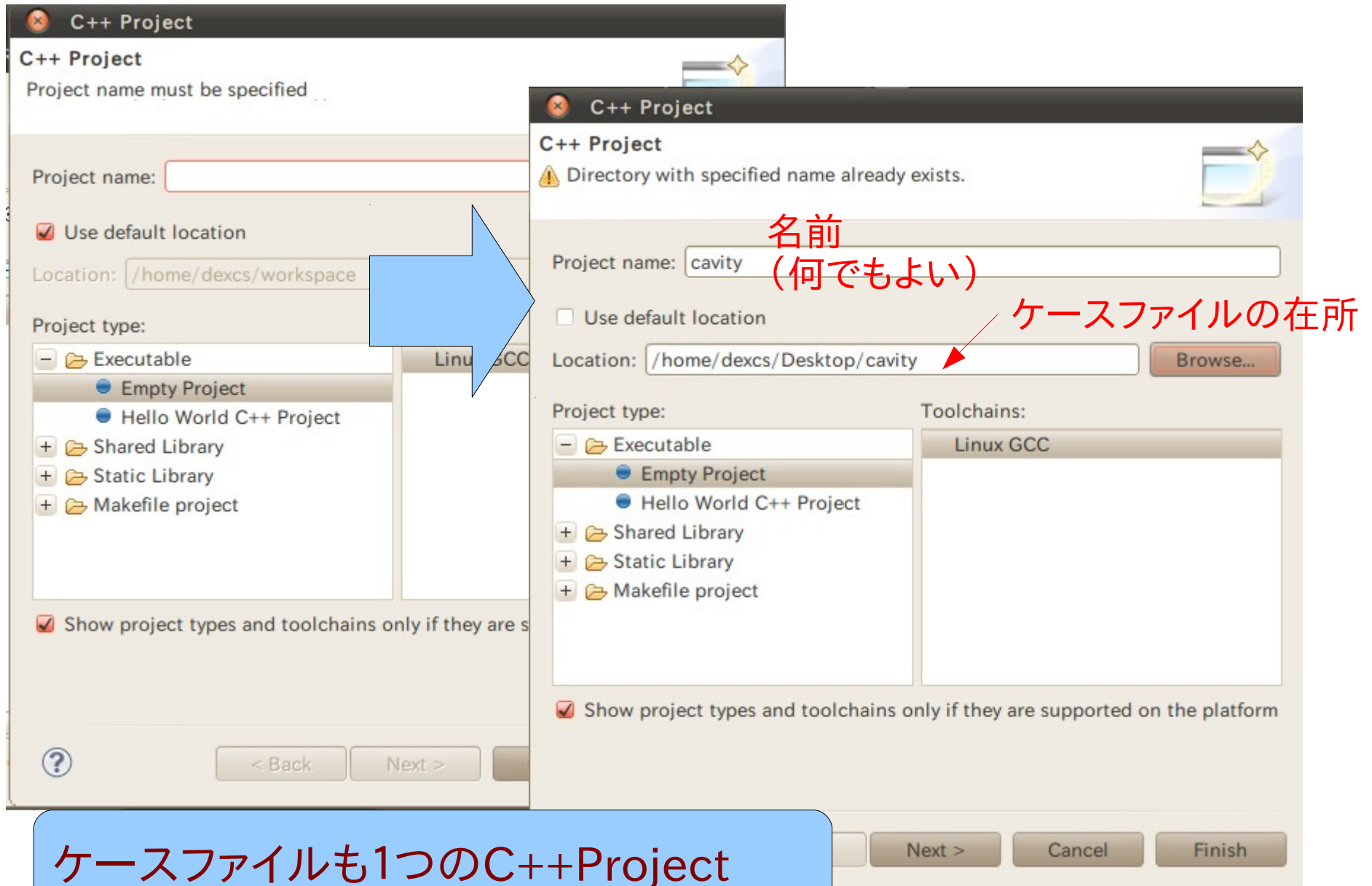
- Deactivate *Use default location* and choose `bubbleFoamMod` so
(*project name*: `bubbleFoamMod`)

ケースファイル⇒C++Project

Files→New→C++Project



C++Projectの設定



Build Location

C++ Project

Select Configurations
Select platforms and configurations you wish to deploy on

Project type: Executable
Toolchains: Linux GCC
Configurations:

- Debug
- Release

Select all
Deselect all

Advanced settings...

Use "Advanced settings" button to edit project's properties. **1**

Additional configurations can be added after project creation.
Use "Manage configurations" buttons either on toolbar or on property pages.

< Back Next > Cancel Finish

Properties for cavity

Resource Builders

- + C/C++ Build
- + C/C++ General

Preferences
Build Settings

C/C++ Build

Configuration: Debug [Active]

Builder Settings Behaviour

Builder
Builder type: External builder

Use default build command

Build command: make Variables...

Makefile generation

Generate Makefiles automatically Expand Env. Variable f

Build location

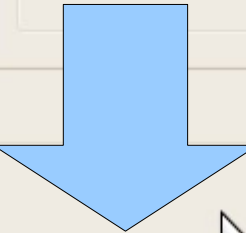
Build directory: \${workspace_loc:/cavity/Debug}

Workspace... File system... Variables...

Restore Defaults Apply

OK Cancel

2



Makefile generation

Generate Makefiles automatically Expand Env. Variable f

Build location

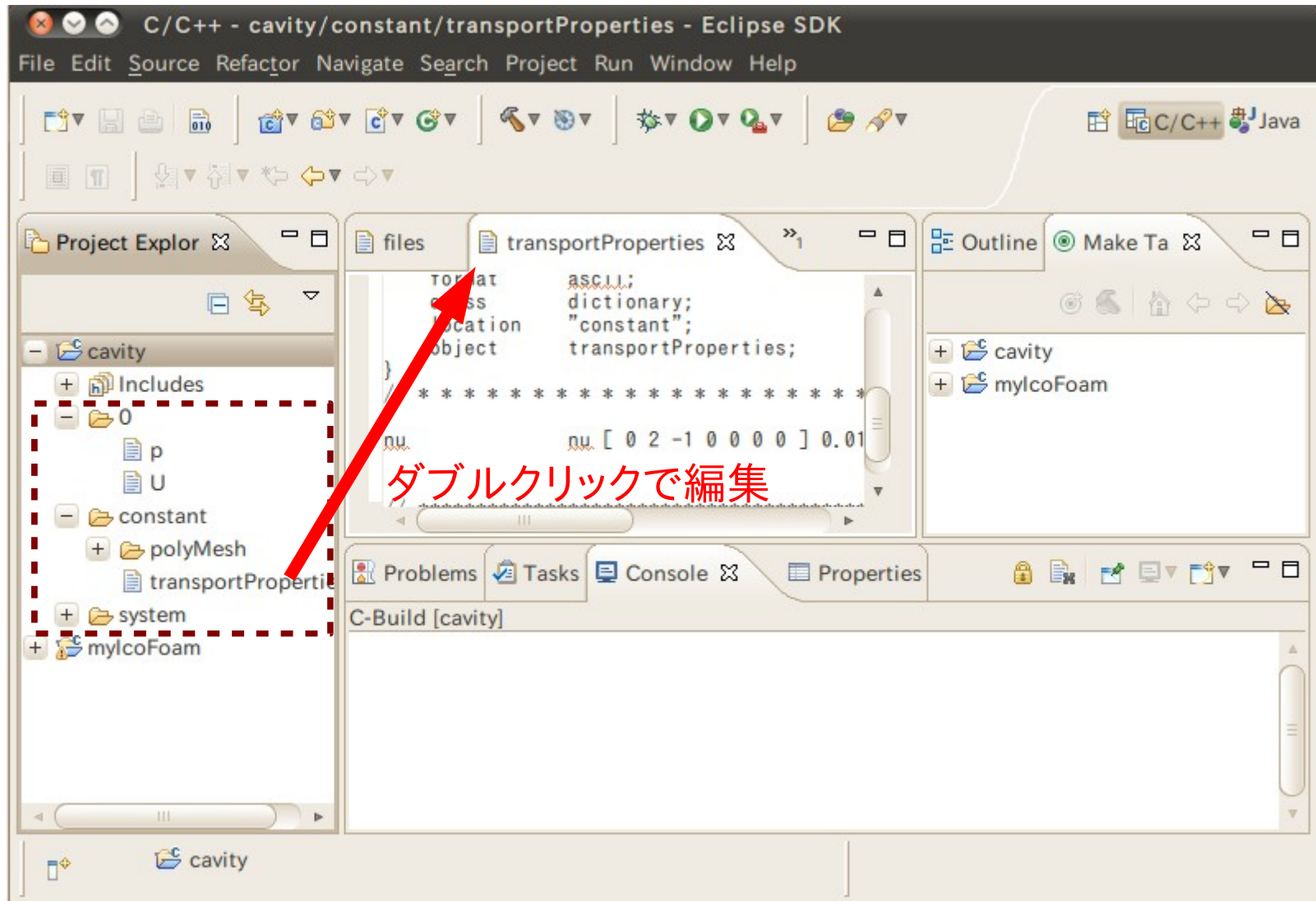
Build directory: /home/dexcs/Desktop/cavity **4**

Workspace... File system... Variables...

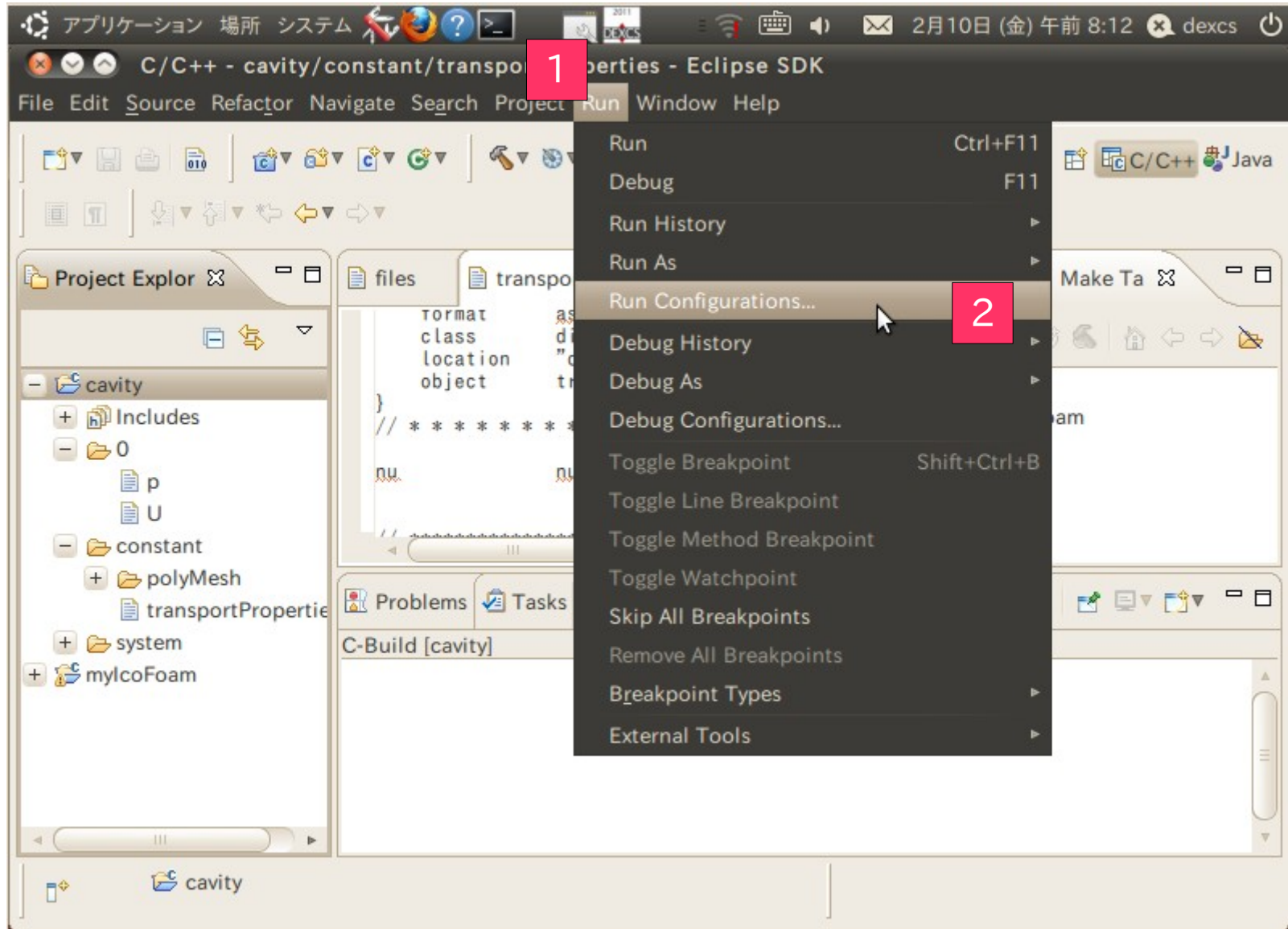
3

Project Location(work directory)
= caseフォルダ

ケースの設定



Run Configurations



Run Configurations

The screenshot shows the Eclipse IDE's Run Configurations dialog. The top window is titled 'Run Configurations' and contains instructions: 'Create, manage, and run configurations', 'Press the 'New' button to create a configuration of the selected type.', 'Press the 'Duplicate' button to copy the selected configuration.', 'Press the 'Delete' button to remove the selected configuration.', and 'Press the 'F' button to filter configurations'. A red box with the number '1' highlights the 'C/C++ Application' option in the left-hand list. A blue arrow points from this list to the main configuration area.

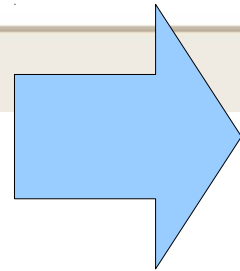
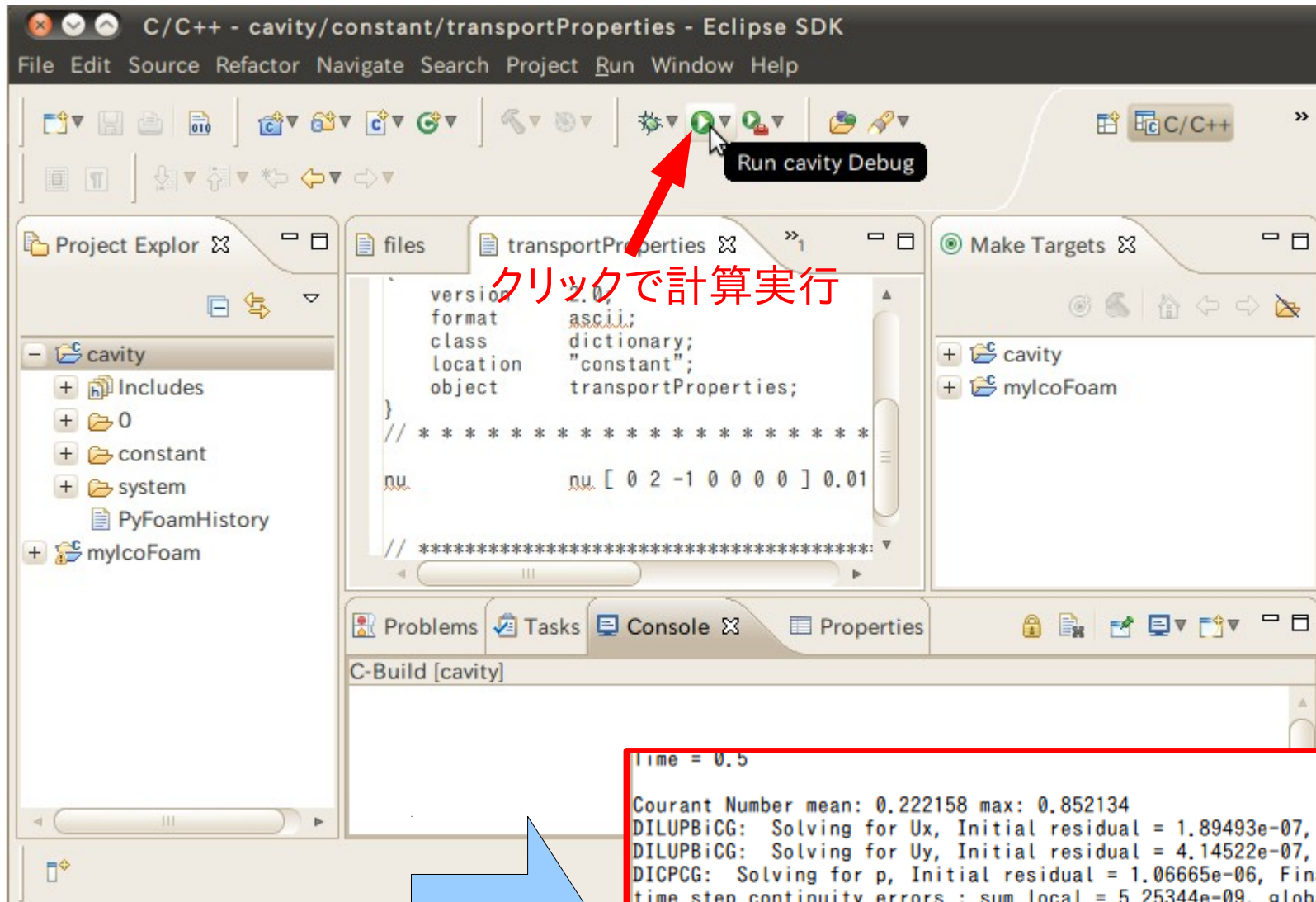
The main configuration area shows a configuration named 'cavity Debug'. The 'Name' field is 'cavity Debug'. The 'Project' is 'cavity'. The 'Build Configuration' is 'Debug'. The 'C/C++ Application' field is '/home/dexcs/OpenFOAM/dexcs-2.0.x/platforms/linux6'. A red box with the number '2' highlights this field. A red arrow points from the 'mylcoFoam' entry in the project explorer to this field.

The project explorer on the right shows the project structure: 'OpenFOAM' (parent), 'OpenFOAM-2.0.x', 'ThirdParty-2.0.x', 'dexcs-2.0.x', 'platforms' (parent), 'linux64GccDPOpt', 'bin' (parent), 'mylcoFoam' (selected, highlighted with a red box and number '3'), 'snappyDictExporter.py', 'run', and 'icoFoam'.

At the bottom, there are buttons for 'Apply', 'Revert', 'Close', and 'Run'.

C/C++ Application
⇒コンパイル済のカスタムソルバを選択

計算実行



```
Time = 0.5  
Courant Number mean: 0.222158 max: 0.852134  
DILUPBiCG: Solving for Ux, Initial residual = 1.89493e-07, Final residual = 1.89493e-07, Cumulative error = 1.89493e-07  
DILUPBiCG: Solving for Uy, Initial residual = 4.14522e-07, Final residual = 4.14522e-07, Cumulative error = 4.14522e-07  
DICPCG: Solving for p, Initial residual = 1.06665e-06, Final residual = 3.39604e-07, Cumulative error = 3.39604e-07  
time step continuity errors : sum local = 5.25344e-09, global = -9.50761e-19, cumulative = -9.50761e-19  
DICPCG: Solving for p, Initial residual = 5.36118e-07, Final residual = 5.36118e-07, Cumulative error = 5.36118e-07  
time step continuity errors : sum local = 6.86432e-09, global = 4.62063e-19, cumulative = 4.62063e-19  
ExecutionTime = 0.15 s ClockTime = 0 s  
  
End
```

計算実行 (別の方法)

1

2

3

4

名前
(何でもよい)

空白

ソルバーの名前

ダブルクリックで
計算実行

Build (コンパイル?) するだけが Make Target ではない

ケースファイル処理の例 (pyFoamClearCase.py)

The screenshot shows the Eclipse IDE interface with the 'Modify Make Target' dialog box open. The dialog has the following fields and options:

- Target name: `clearCase` (Annotated with red text: 名前 (何でもよい))
- Make Target: Same as the target name
- Make target: `.` (Annotated with red text: ブランク)
- Build Command: Use builder settings
- Build command: `pyFoamClearCase.py` (Annotated with red text: 初期化コマンド)
- Build Settings: Stop on first build error, Run all project builders

Annotations in the image:

- 1: Points to the 'Run' button in the dialog.
- 2: A bracket grouping the 'Target name', 'Make target', and 'Build command' fields.
- 3: Points to the 'OK' button at the bottom of the dialog.
- 4: Points to the 'clearCase' target in the Outline view.
- 5: Points to the 'clearCase' target in the Project Explorer.

Make Target ⇒ コマンド登録 と読み替えば、何かと工夫出来そう...

Eclipse上でOpenFOAMを使う・・・とは

- ソースコードの参照(勉強)
- カスタムソルバーのコンパイル
- 上記カスタムソルバーでケースファイルを実行
- 同上、デバッグ

デバッグ

```
icoFoam.C ✕
62
63 solve(UEqn == -fvc::grad(p));
64
65 // --- PISO loop
66
67 for (int corr=0; corr<nCorr; corr++)
68 {
69     volScalarField rAU(1.0/UEqn.A());
70
71     U = rAU*UEqn.H();
72     phi = (fvc::interpolate(U) & mesh.Sf())
73           + fvc::ddtPhiCorr(rAU, U, phi);
74
75     adjustPhi(phi, U, p);
76
77     for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
78     {
79         fvScalarMatrix pEqn
80         (
81             fvm::laplacian(rAU, p) == fvc::div(phi)
82         );
83
84         pEqn.setReference(pRefCell, pRefValue);
85         pEqn.solve();
86
87         if (nonOrth == nNonOrthCorr)
88         {
```

```
*icoFoam.C ✕
62
63 solve(UEqn == -fvc::grad(p));
64
65 // --- PISO loop
66
67 for (int corr=0; corr<nCorr; corr++)
68 {
69     volScalarField rAU(1.0/UEqn.A());
70
71     U = rAU*UEqn.H();
72     phi = (fvc::interpolate(U) & mesh.Sf())
73           + fvc::ddtPhiCorr(rAU, U, phi);
74
75     adjustPhi(phi, U, p);
76
77     for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
78     {
79         fvScalarMatrix pEqn
80         (
81             fvm::laplacian(rAU, p) == fvc::div(phi)
82         );
83
84         Info<< "nonOrth = " << nonOrth << endl;
85
86         pEqn.setReference(pRefCell, pRefValue);
87         pEqn.solve();
88
89         if (nonOrth == nNonOrthCorr)
90         {
```

不まじめプログラマの作法

デバッグ gdb

The screenshot shows the Synaptic Package Manager window. The title bar reads "Synaptic パッケージ・マネージャ". The menu bar includes "ファイル(F)", "編集(E)", "パッケージ(P)", "設定(S)", and "ヘルプ(H)". The toolbar contains icons for "再読み込み", "すべてアップグレード", "適用", "プロパティ", and a search icon labeled "検索".

The main window is divided into two panes. The left pane shows a list of categories under "すべて", including "GNOMEデスクトップ環境", "GNU R言語 (statistical system)", "Gnustep デスクトップ環境 (univ)", "Haskell プログラミング言語 (un)", "Java プログラミング言語", and "Java プログラミング言語 (univer)". Below this list are buttons for "セクション(S)", "状態(T)", "ダウンロード元", "カスタムフィルタ(C)", and "検索結果(E)".

The right pane displays a table of installed and available packages. The table has columns for "S", "パッケージ", "インストール済バージョン", and "最新バージョン". The "gdb" package is highlighted in red, indicating it is installed. The table shows the following data:

S	パッケージ	インストール済バージョン	最新バージョン
☆	gdb		7.11-ubuntu2
☆	gdb-doc		7.11-1
☆	gdb-avr		6.4.90.dfsg-3u
☆	gdal-bin		1.6.3-3build2
☆	gdcursor		0.061-ubuntu
☆	gdb	7.1-1ubuntu2	7.1-1ubuntu2

Below the table, there is a section titled "GNU デバッガ" with a button "スクリーンショットを見る". The text describes GDB as a source-level debugger and lists supported languages: "現在、C, C++, Fortran, Modula 2 および Java プログラムに対して動作します。まじめな全てのプログラマにとっての必需品です。". A red underline is present under the last sentence. Below this, it states: "Canonical は gdb の重要なアップデートを 2015年4月 まで提供します。".

At the bottom of the window, a status bar reads: "全 30915 パッケージ (インストール済: 1504 個, 破損 0 個, インストールまたはアップグレード指定: 0 個, 削除: 0 個)".

gdb を用いたデバッグ方法

<http://uguisu.skr.jp/Windows/gdb.html>

gdb を利用する

ソースコードが大量にあり、どのソースコードでエラーが発生しているか分からない場合には威力を発揮します。

gdb を利用する

```
$ gcc -g ソースコード名
```

-g オプションによってデバッグのための情報分、実行ファイルのサイズが大きくなります。

起動方法

```
$ gdb 実行ファイル名
```

表示(変数の出力)方法

```
(gdb) p 変数表示  
(gdb) p *ポインタ変数  
(gdb) p &アドレス変数
```

p は print の省略です。

print コマンドに /format を付加することで、出力書式を指定することができます。

o	8進表示	d	符号付き10進表示
x	16進表示	u	符号なし10進表示
t	2進表示	c	文字表示
f	浮動小数点表示	a	アドレス

ブレークポイントを指定する

ブレークポイントを設定すると、そこで実行が一時停止します。

```
(gdb) b 行番号  
(gdb) b 関数名  
(gdb) b ファイル名:行番号
```

b は break の省略です。

ジャンプ

関数の中の処理まで追い掛けたくない場合。

```
(gdb) n
```

n は next の省略です。

関数内の処理を追いかける場合。

```
(gdb) s
```

s は step の省略です。

ブレークポイントを設定した箇所だが、実行の継続する場合。

基本はコマンドラインツールだが
Eclipse上でGUI制御可能

gdbのバージョン問題

GNU Debugger (gdb) version: 6.8

Please be careful choosing the right gdb version! In gdb 7.x a bug seems to occur while debugging and monitoring the variables.

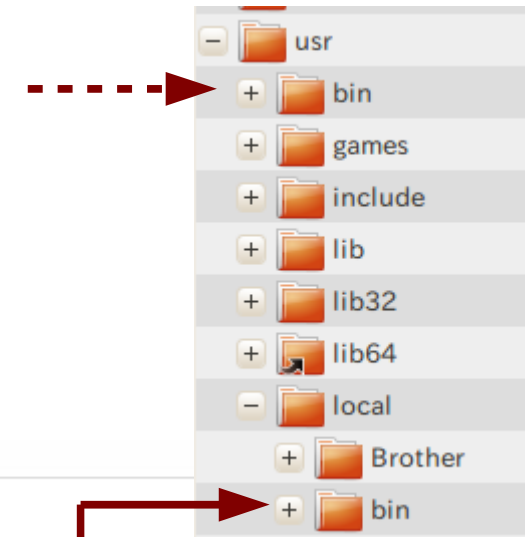
<http://mogura7.zenno.info/~et/xoops/modules/wordpress/index.php?p=444>

gdbのアンインストール

- gdbはデバッガですが、Eclipseでデバッグする場合、ubuntuに標準搭載されたgdbでは正しく動作しない為、ついでにSynapticパッケージ・マネージャ上で、これのアンインストールをやっておきます。



不要



- 全 30832 パッケージ (インストール済: 1567 個, 破損 0 個, インストールまたはアップグレード済: 0 個, 削除: 0 個)
- gdb(緑色のマークが付いてインストール済)を選択して、マウスボタン右クリック⇒「削除指定」を選択

gdbのインストール

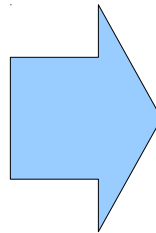
- gdbは、<http://ftp.gnu.org/gnu/gdb/> から、旧版(gdb-6.8)をダウンロードする。
- ダウンロードしたgdb-6.8a.tar.gzを適当な場所(ここではデスクトップ上)で展開し、通常の端末画面を開き、以下入力

```
cd Desktop/gdb-6.8
./configure --disable-werror
make
sudo make install
```

- ちなみにconfigureのオプション(--disable-werror)を指定しないと、makeで失敗します。READMEには書いてなくて、google検索でも、gdb-6.8でコンパイルが出来ないという記事はたくさん見つかるものの、この解決策に辿りつくまでは相当な難航でした。

gdb-6.8 のインストール

```
README ✕
9 package -- supported targets, how to use
10
11 It is now possible to automatically conf
12 tools with one command. To build all of
13 run the ``configure'' script here, e.g.:
14
15     ./configure
16     make
17
18 To install them (by default in /usr/loca
19 then do:
20     make install
21
```



```
et@et-Raytrek64: ~/Desktop/gdb-6.8
ファイル(F) 編集(E) 表示(V) 端末(T) ヘルプ(H)
CONFIG_COMMANDS=gdb_stdint.h ¥
CONFIG_FILES= ¥
CONFIG_HEADERS= ¥
CONFIG_LINKS= ¥
/bin/bash config.status
config.status: executing gdb_stdint.h commands
echo stamp > stamp-int
gcc -c -g -O2 -I. -I../gdb -I../gdb/config -DLOCALEDIR="¥"/usr/local/share
/locale¥" -DHAVE_CONFIG_H -I../gdb/./include/opcode -I../gdb/./readline/.
.. -I../bfd -I../gdb/./bfd -I../gdb/./include -I../libdecnumber -I../gdb/
../libdecnumber -DMI_OUT=1 -DTUI=1 -Wall -Wdeclaration-after-statement -Wpoin
ter-arith -Wformat-nonliteral -Wno-pointer-sign -Wno-unused -Wno-switch -Wno-cha
r-subscripts -Werror linux-nat.c
cc1: warnings being treated as errors
linux-nat.c: In function 'linux_nat_info_proc_cmd':
linux-nat.c:2879: error: ignoring return value of 'fgets', declared with attribu
te warn_unused_result
make[2]: *** [linux-nat.o] エラー 1
make[2]: ディレクトリ `~/home/et/Desktop/gdb-6.8/gdb' から出ます
make[1]: *** [all-gdb] エラー 2
make[1]: ディレクトリ `~/home/et/Desktop/gdb-6.8' から出ます
make: *** [all] エラー 2
et@et-Raytrek64:~/Desktop/gdb-6.8$ gedit README
et@et-Raytrek64:~/Desktop/gdb-6.8$
```

Error while installing gdb

<http://ubuntuforums.org/showthread.php?t=1009449>

December 12th, 2008

#1

joker@

First Cup of Ubuntu



Join Date: Dec 2008

Beans: 6

erros while installing GDB

Hi

I trying install GDB on my U 8.10

this is what i get after send command 'make" its polish language but is simple what is where,any sugiestion??

regards

```
make[2]: Nie ma nic do zrobienia w `all'.
```

```
make[2]: Opuszczenie katalogu `/home/rav/Pulpit/gdb-6.8/libdecnumber'
```

```
make[2]: Wejście do katalogu `/home/rav/Pulpit/gdb-6.8/readline'
```

```
make[2]: Nie ma nic do zrobienia w `all'.
```

```
make[2]: Opuszczenie katalogu `/home/rav/Pulpit/gdb-6.8/readline'
```

```
make[2]: Wejście do katalogu `/home/rav/Pulpit/gdb-6.8/sim'
```

```
make[2]: Opuszczenie katalogu `/home/rav/Pulpit/gdb-6.8/sim'
```

```
make[2]: Wejście do katalogu `/home/rav/Pulpit/gdb-6.8/gdb'
```

```
gcc -c -g -O2 -I. -I../gdb -I../gdb/config -DLOCALEDIR="" -DHAVE_CONFIG_H -I../gdb/./include/opcode -I../gdb/./readline/.. -I../bfd -I../gdb/./bfd -I../gdb/./include -I../libdecnumber -I../gdb/./libdecnumber -DMI_OUT=1 -DTUI=1 -Wall -Wdeclaration-after-statement -Wpointer-arith -Wformat-nonliteral -Wno-pointer-sign -Wno-unused -Wno-switch -Wno-char-subscripts -Werror linux-nat.c
```

```
cc1: warnings being treated as errors
```

July 4th, 2010

#6

PLayer_unu

First Cup of Ubuntu



Join Date: Jul 2010

Location: Craiova, Romania

Beans: 1

Re: I got similar errors compiling GDB

Quote:

Originally Posted by skierpage

*I had this and several similar compiler errors trying to compile gdb 6.8 under Kubuntu 9.04 64-bit. I filed bugs [10674](#) and [10675](#) against gdb at sourceware.org. You can get past them by declaring a junk variable of the right type, e.g. `char *junk;` at the top of the program block, then assigning `junk = [the problem function call]`.*

I compiled gdb myself hoping to avoid gdb crashes in `iterate_over_threads`, possibly because Ubuntu should have repackaged gdb for kernel and/or libc updates since 9.04, which might be [Launchpad bug 258578](#). But it didn't help.

It's a good solution to declare a junk variabik
another FAST solution.

Edit the Makefile and delete this line:

~~WERROR_CFLAGS = -Werror~~

That's because you will be prompted for an e



red, but then I've got another 5-6 similar errors and I've found

compiler will get a warning.



Eclipseでデバッグ

Debugging with Eclipse

参考資料1

Access to runTime-selective models

Set up debug configurations

- *Run* → *Debug Configurations* – Choose same configurations as for run
- Check the use of GDB (DSF) Create Process Launcher as debugger
- Click the *Debug* button

Set breakpoints

- 1st breakpoint: `createFields.H`
`autoPtr<dragModel> draga = dragModel::New(..);`

Process Launcher

Debug Configurations
Create, manage, and run configurations
Multiple launchers available - select one to continue

Name: cavity Debug

Project: cavity

C/C++ Application: /opt/OpenFOAM/dexcs-2.0.x/platforms/linux64GccD

Select Preferred Launcher
This dialog allows you to specify which launcher to use when multiple launchers are available for a configuration and launch mode.
 Use configuration specific settings [Change Workspace Settings...](#)

Launchers:

- GDB (DSF) Create Process Launcher
- GDB (DSF) Remote System Process Launcher
- Standard Create Process Launcher

Description
Start new application under control of GDB debugger integrated using the Debugger Services Framework (DSF).

Buttons: Cancel, OK, Apply, Revert, Close, Debug

1: Multiple launchers available - [Select one...](#)

2: GDB (DSF) Create Process Launcher

3: OK

4: Debug

C/C++ - mylcoFoam/createFields.H - Eclipse SDK

File Edit Source Refactor Navigate Search Run Project Window Help

Project Explorer: cavity, mylcoFoam

Source Editor: createFields.H

```
IOobject
(
    "transportProperties",
    runTime.constant(),
    mesh,
    IOobject::MUST_READ_IF_MODIFIED,
    IOobject::NO_WRITE
);

dimensionedScalar nu
(
    transportProperties.lookup("nu")
);
```

Breakpoint attribute problem: installation failed

unused variable 'momentumPredictor'

デバッグ用 コンパイルオプション

参考資料3

- 1 Thanks to ..
- 2 Versions
- 3 Eclipse Capabilities
- 4 Aim of the tutorial
- 5 Download and set up Eclipse
 - 5.1 Install CDT by hand
 - 5.2 Launching C++ project
- 6 Developing and compiling
- 7 Running applications and debugging
 - 7.1 Debugging in Parallel
- 8 Profiling with gProf and Linux Tools

It is also possible to debug applications under Eclipse with the GNU Debugger gdb. Therefore you may need to compile your OpenFOAM version in the debug mode of the compiler. Change the compiler settings in *OpenFOAM/OpenFOAM-1.6.x/etc/bashrc* to debug with *WM_COMPILE_OPTION:=Debug* and compile OpenFOAM using *./Allwmake*. This will take a few minutes.

gdb を利用する

ソースコードが大量にあり、どのソースコードでエラーが発生しているか分からない場合には威力を発揮します。

gdb を利用する

```
$ gcc -g ソースコード名
```

-g オプションによってデバッグのための情報分、実行ファイルのサイズが大きくなります。

起動方法

```
$ gdb 実行ファイル名
```

- 10 Groups on extend-project
- 11 Training Session from 6th OpenFOAM Wor
Eclipse for OpenFOAM

OpenFOAM環境変数

デバッグ用

```
bashrc ✕  
76  
77 #- Optimised, debug, profiling:  
78 #   WM_COMPILE_OPTION = Opt | Debug | Prof  
79 export WM_COMPILE_OPTION=Debug  
80
```

-	OpenFOAM	4個のアイテム
-	OpenFOAM-2.0.x	12個のアイテム
+	applications	4個のアイテム
+	bin	60個のアイテム
+	doc	6個のアイテム
+	etc	7個のアイテム
-	platforms	2個のアイテム
-	linux64GccDPDebug	2個のアイテム
+	bin	243個のアイテム
+	lib	94個のアイテム
-	linux64GccDPOpt	2個のアイテム
+	bin	248個のアイテム
+	lib	102個のアイテム
+	src	31個のアイテム
+	tutorials	16個のアイテム
+	wmake	22個のアイテム
	Allwmake	923 バイト

通常

```
bashrc ✕  
76  
77 #- Optimised, debug, profiling:  
78 #   WM_COMPILE_OPTION = Opt | Debug | Prof  
79 export WM_COMPILE_OPTION=Opt  
80
```

It is also possible to debug applications under Eclipse with
Therefore you may need to compile your OpenFOAM version
compiler. Change the compiler set
debug with WM_COMPILE_OPTION
This will take a few minutes.



カスタムソルバ の再コンパイル

①Project選択

②wcleanをダブルクリック

③wmakeをダブルクリック

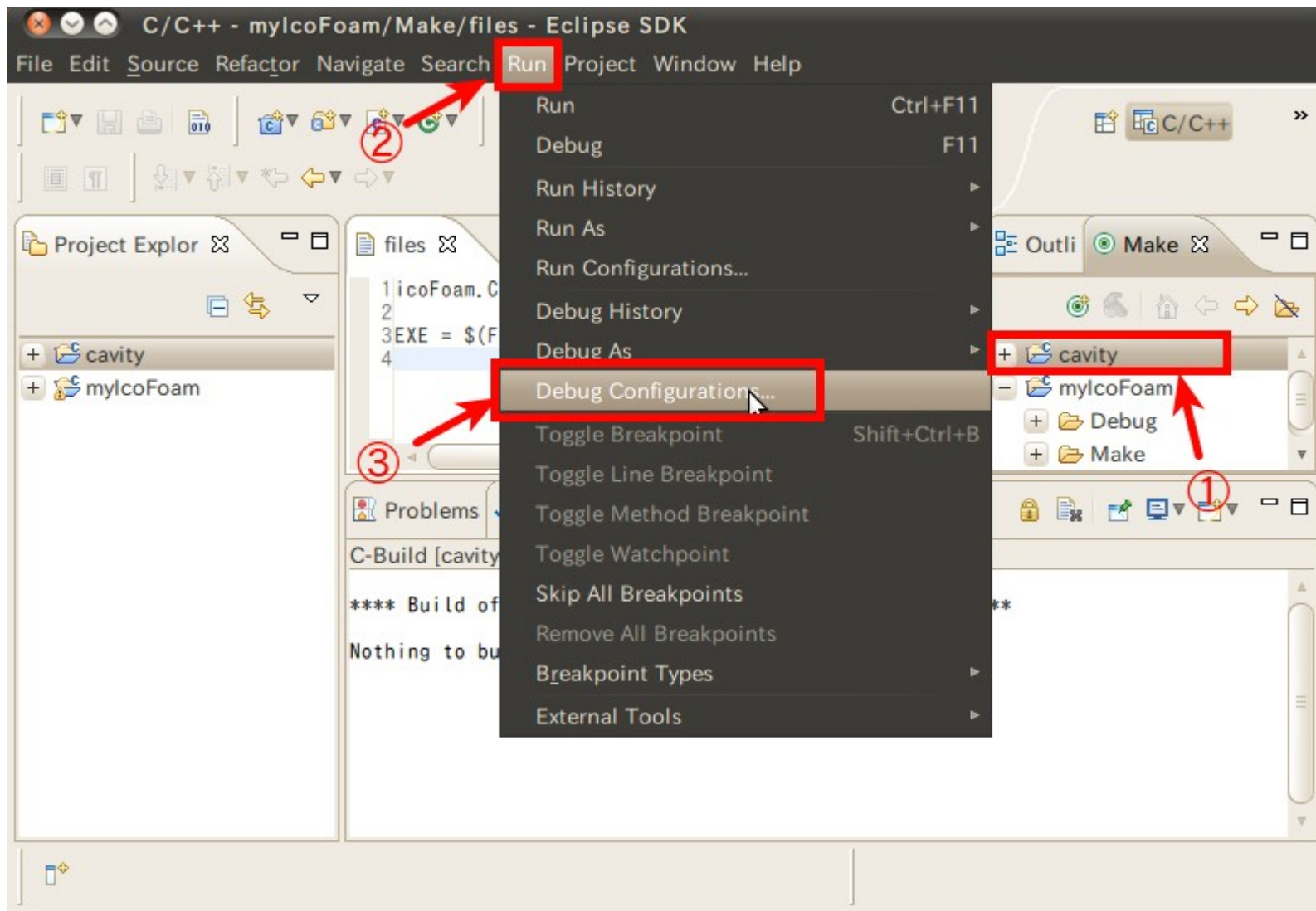
④Debug フォルダに作成された

```
icoFoam.C
EXE = $(FOAM_USER_APPBIN)/myIcoFoam

C-Build [myIcoFoam]
/home/dexcs/openfoam/OpenFOAM-2.0.x/src/finiteVolume/finiteVolume/
readPISOControls.H:15: warning: unused variable 'transonic'
g++ -m64 -Dlinux64 -DWM_DP -Wall -Wextra -Wno-unused-parameter -Wold-style-cast
-Wnon-virtual-dtor -O0 -fdefault-inline -ggdb3 -DFULLDEBUG -DNoRepository -
ftemplate-depth-100 -I/home/dexcs/OpenFOAM/OpenFOAM-2.0.x/src/finiteVolume/
lnInclude -IlnInclude -I. -I/home/dexcs/OpenFOAM/OpenFOAM-2.0.x/src/OpenFOAM/
lnInclude -I/home/dexcs/OpenFOAM/OpenFOAM-2.0.x/src/OSspecific/POSTX/lnInclude
-fPIC -Xlinker --add-needed Make/linux64GccDPDebug/icoFoam.o -L/home/dexcs/
OpenFOAM/OpenFOAM-2.0.x/platforms/linux64GccDPDebug/lib ¥
-lfiniteVolume -lOpenFOAM -ldl -ggdb3 -DFULLDEBUG -lm -o /home/dexcs/
OpenFOAM/dexcs-2.0.x/platforms/linux64GccDPDebug/bin/myIcoFoam
```

-	OpenFOAM	4個のアイテム
+	OpenFOAM-2.0.x	12個のアイテム
+	ThirdParty-2.0.x	17個のアイテム
-	dexcs-2.0.x	2個のアイテム
-	platforms	2個のアイテム
-	linux64GccDPDebug	1個のアイテム
-	bin	1個のアイテム
	myIcoFoam	2.6 MB
-	linux64GccDPOpt	1個のアイテム
-	bin	2個のアイテム
	myIcoFoam	480.9 KB
	snappyDictEx...	19.8 KB
+	run	1個のアイテム

Debug Configurations



Debug Configurations

The image shows the Eclipse IDE's 'Debug Configurations' dialog box and a file explorer window. The dialog box is titled 'Debug Configurations' and has the subtitle 'Create, manage, and run configurations'. It features a list of configuration types on the left, with 'C/C++ Application' selected. The main area shows a configuration named 'cavity Debug' with the following details:

- Name: cavity Debug
- Project: cavity
- C/C++ Application: /opt/OpenFOAM/dexcs-2.0.x/platforms/linux64GccDP... (with a 'Browse...' button)

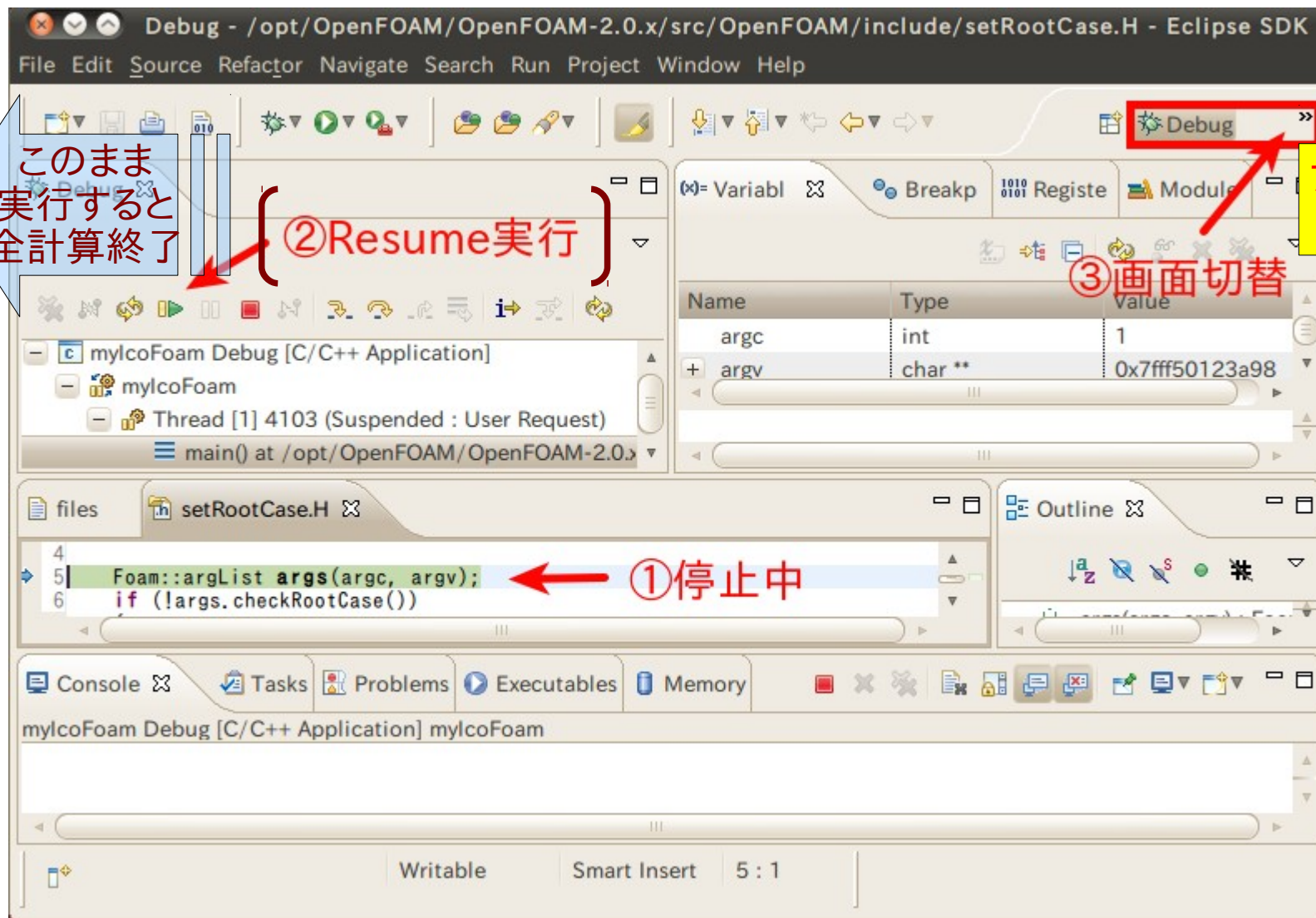
The file explorer window shows a directory structure for 'OpenFOAM'. A red box highlights the 'mylcoFoam' file in the 'bin' directory of the 'linux64GccDPOpt' folder. A red arrow labeled '1' points from this file to the 'Browse...' button in the dialog. Another red arrow labeled '2' points from the 'cavity Debug' configuration in the dialog to the 'mylcoFoam' file. A third red arrow labeled '3' points from the 'Debug' button at the bottom of the dialog to a blue arrow pointing to the text 'デバッガの起動' (Start Debugger).

Using GDB (DSF) Create Process Launcher - [Select other...](#) Apply Revert

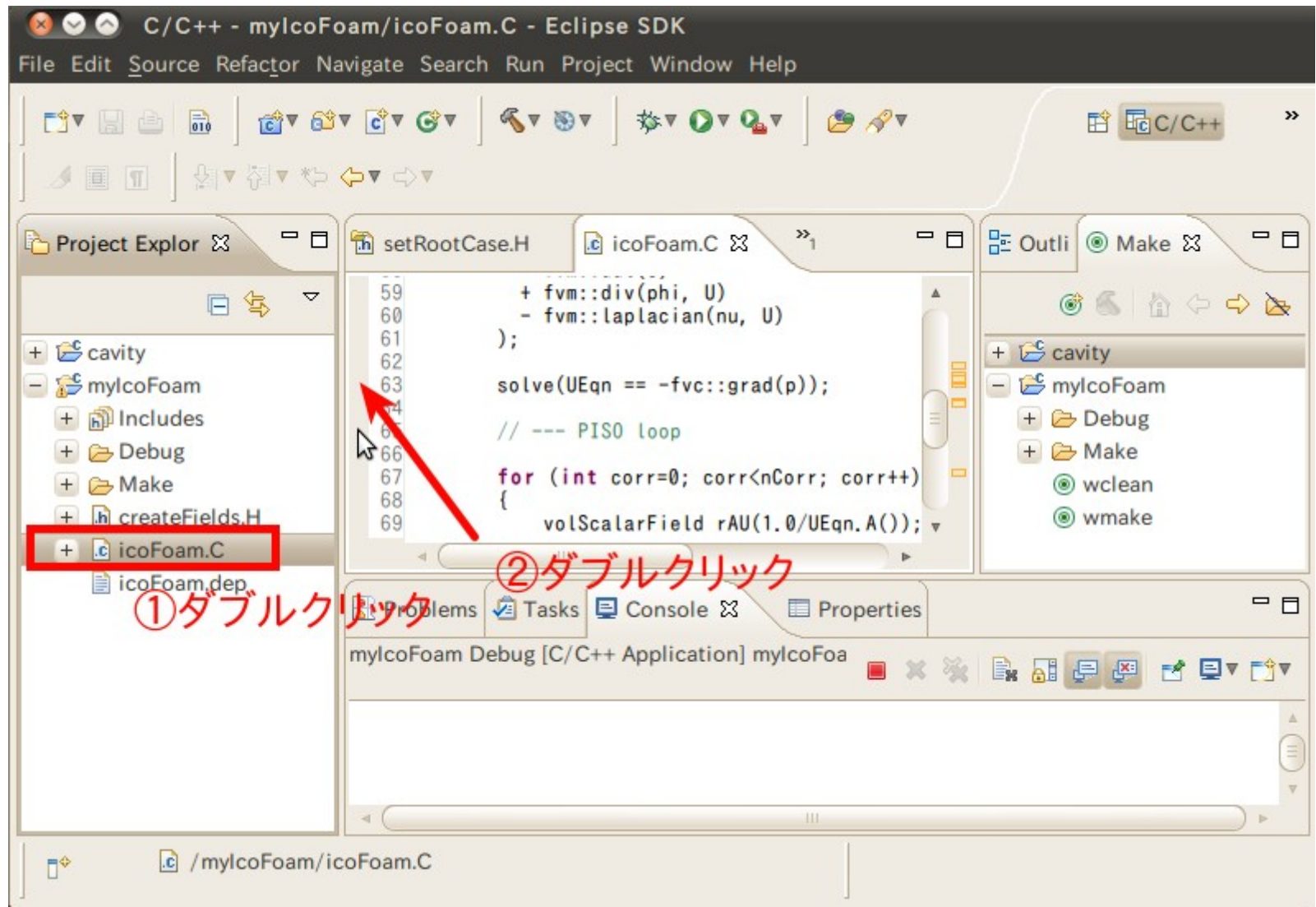
Close Debug

デバッガの起動

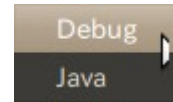
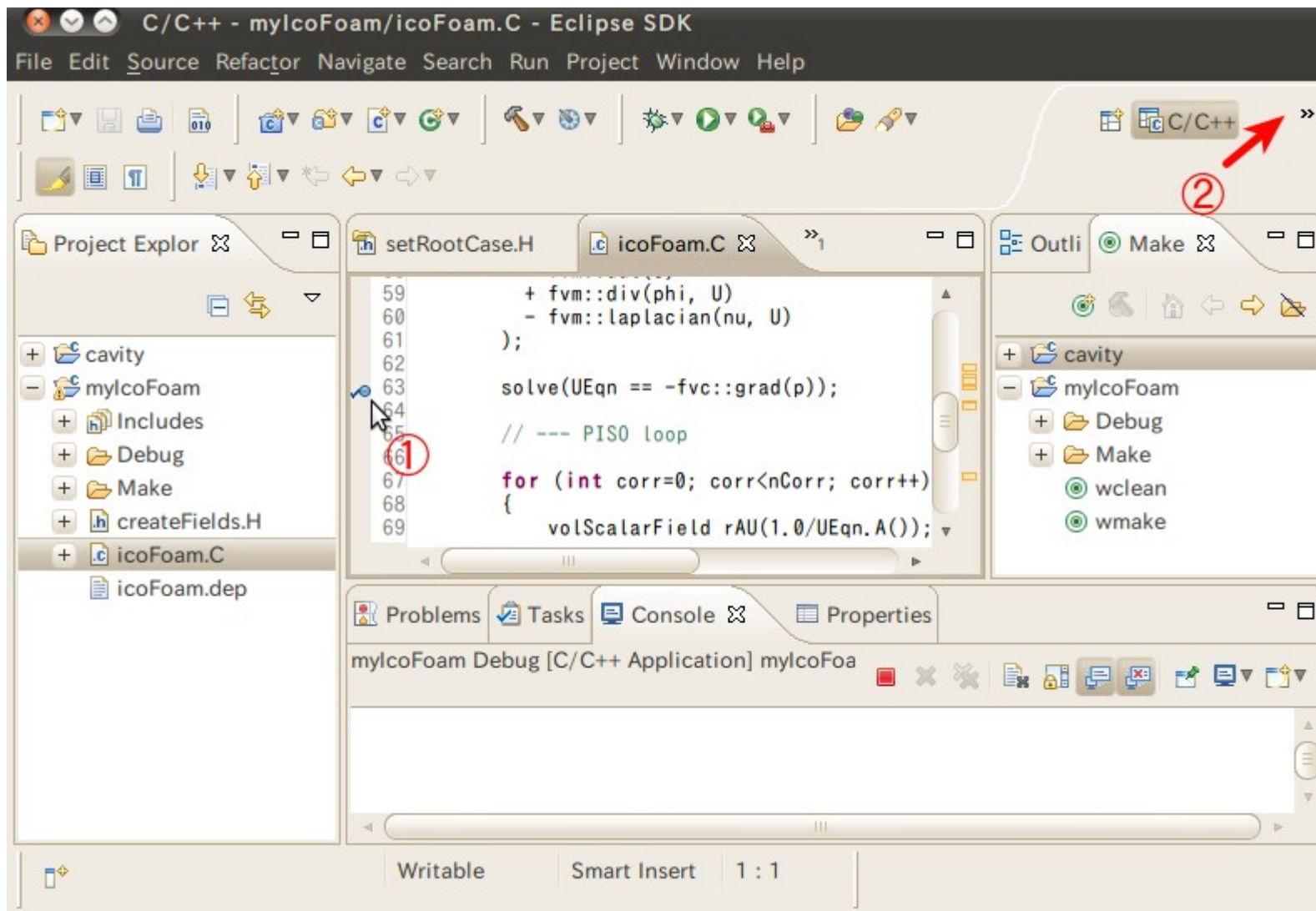
デバッガの起動



ブレークポイントの設定

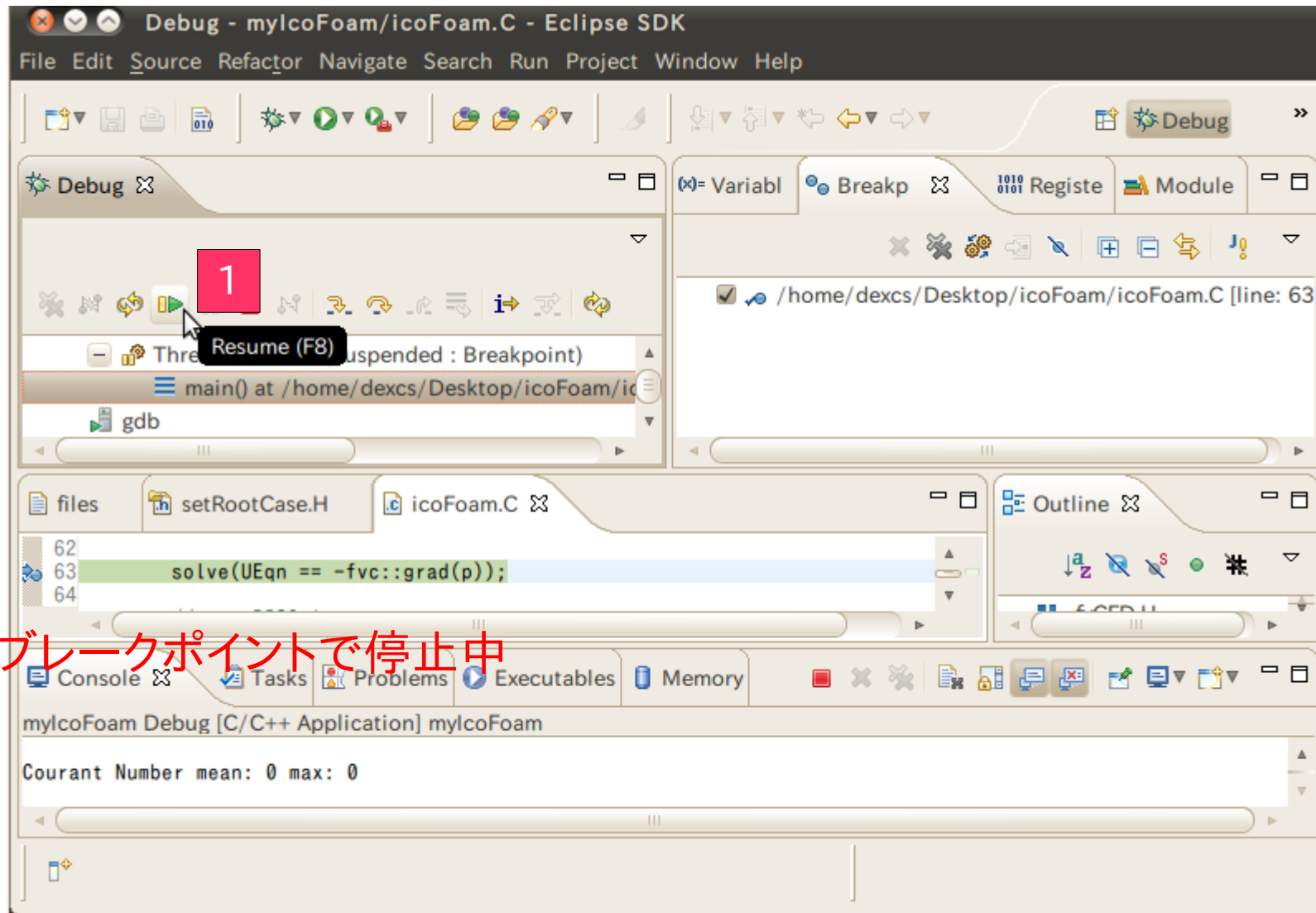


ブレークポイント設定完了



デバッガ
のResume

デバッガのResume



ブレークポイントで停止中

デバッガで出来ること

再度Resume

変数の内容を確認可能

Name	Type	Value
+ mesh	class Foam::fvMesh	[...]
- nu	dimensionedScalar	[...]
+ name_	Foam::word	[...]

3

Time = 0.01 ← Timeループが1つ進行!

Courant Number mean: 0.0976805 max: 0.585722