

## 物理シミュレーション (Bullet Physics) の使用方法

東 俊光

### 1. Bullet Physics 環境の構築

#### (1) Bullet Physics のダウンロード

Bullet Physics の最新版をダウンロードする.

- 下記のサイトにリストが表示されている. 最新版の圧縮ファイルをダウンロードする.

(参考) <https://code.google.com/p/bullet/downloads/list>

- MacOSX 上で動作する Bullet Physics 環境は, zip ファイルをダウンロードすると作成可能である.

- 2014 年 7 月 29 日現在の最新版は bullet-2.82-r2704 となる.

#### (2) ダウンロードした圧縮ファイルを回答する. 回答するアプリケーションは特に問わないが, 日本語文字のエンコードが正しく行われることを考慮すると, Stuffit Expander が望ましい.

#### (3) ファイルを回答したフォルダの最上位 (解凍フォルダのルートフォルダ) に移動する.

フォルダに移動後, 下記のコマンドを入力する.

※ ただし “¥” は “(バックスラッシュ)” とする必要がある.

```
cmake . -G "Unix Makefiles" -DINSTALL_LIBS=ON
-DDBUILD_SHARED_LIBS=ON -DFRAMEWORK=ON ¥
-DMAKE_OSX_ARCHITECTURES='i386;x86_64'
-DMAKE_BUILD_TYPE=RelWithDebInfo ¥
-DMAKE_INSTALL_PREFIX=/Library/Frameworks
-DMAKE_INSTALL_NAME_DIR=/Library/Frameworks ¥
-DBUILD_DEMOS:BOOL=OFF
```

#### (4) 上記コマンドを入力した場合は, ダウンロードされたディレクトリ内の下記にインストールされる.

- (Bullet Physics のルートディレクトリ)/src/BulletCollision
- (Bullet Physics のルートディレクトリ)/src/BulletDynamics
- (Bullet Physics のルートディレクトリ)/src/LinearMath

#### (5) 上記のコマンドは framework を Xcode に取り込む方法を表したものであるが, /usr/local/include 及び /usr/local/lib にインストールする方法もあるの

で注意が必要.

(6) 参考となるWebサイト

<http://www.bulletphysics.org/mediawiki-1.5.8/index.php?title=Installation>

2. デモプログラムの作成

(1) Bullet Physics の潤沢なライブラリを使用するには, Computer Science の専門でも無い限り, 予備知識, 事前 to 取得した環境無しで使いこなすのは難しいと考える.

(2) Bullet Physics の特色は, 豊富に作成された (但し分かりにくい) デモプログラムにある. このデモプログラムを作成する中で, 様々な関数, クラスを作成している. 我々が自らのシミュレーションプログラムを作成する上で, これを使用しない手は無い.

(3) 以上から, できる限り簡便に Bullet Physics の環境を使用するのであれば, 下記のディレクトリをインクルードディレクトリに取り込むと良い.

- (Bullet Physics ルートディレクトリ)/src ※framework 取込時の設定
- (Bullet Physics ルートディレクトリ)/Demos/OpenGL (具体例)

- /Users/east/programming/bullet-2.82-r2704/src

- /Users/east/programming/bullet-2.82-r2704/Demos/OpenGL

※ 上記はいずれも non-recursive 設定となる.

(4) 他に可能な設定では, インクルードファイルは下記の設定にする.

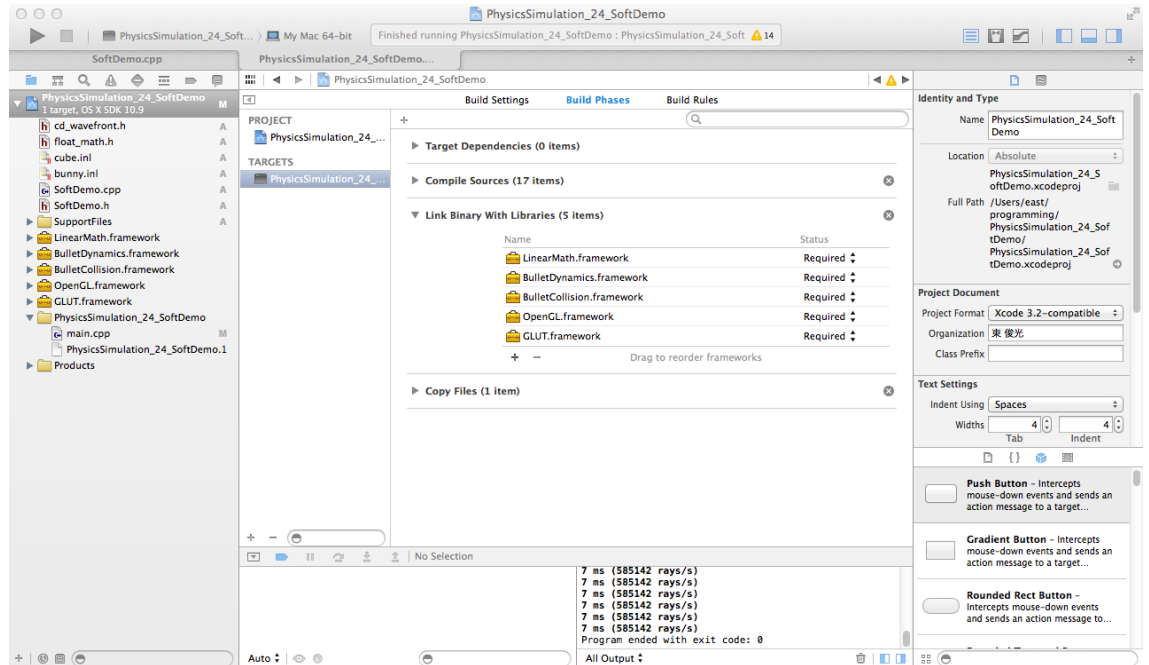
- (Bullet Physics ルートディレクトリ)/src/BulletCollision
- (Bullet Physics ルートディレクトリ)/src/BulletDynamics
- (Bullet Physics ルートディレクトリ)/src/LenearMath

※ 上記はいずれも non-recursive 設定となる.

(5) framework を取り込む. framework とは動的共有ライブラリ, nib ファイル, image ファイル, ローカライズファイル, ヘッダファイル, ドキュメント等のリソースファイルを 1 箇所にまとめたディレクトリを指す. これにより, ヘッダファイルをライブラリとまとめて配布できるので使い勝手が良い. 例えば開発時, 静的ライブラリ (\*.a) 形式だと他にヘッダファイルも配置する必要があり, ファイルが増えると煩雑になる.

2014 年 7 月 29 日

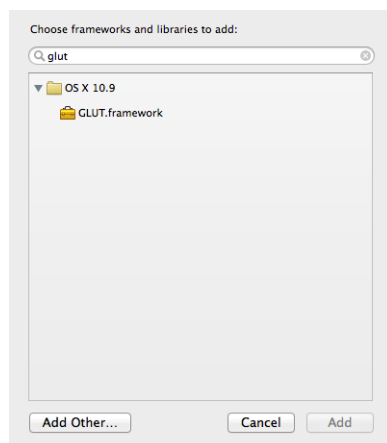
- (6) framework の追加は「TARGET」を選択し、「Build Phases」上の設定を選択することで現れる「+」マークをクリックすることで可能となる。



- (7) Bullet Physics の環境を整えるためには、下記の 5 つの framework を取り込むと良い。

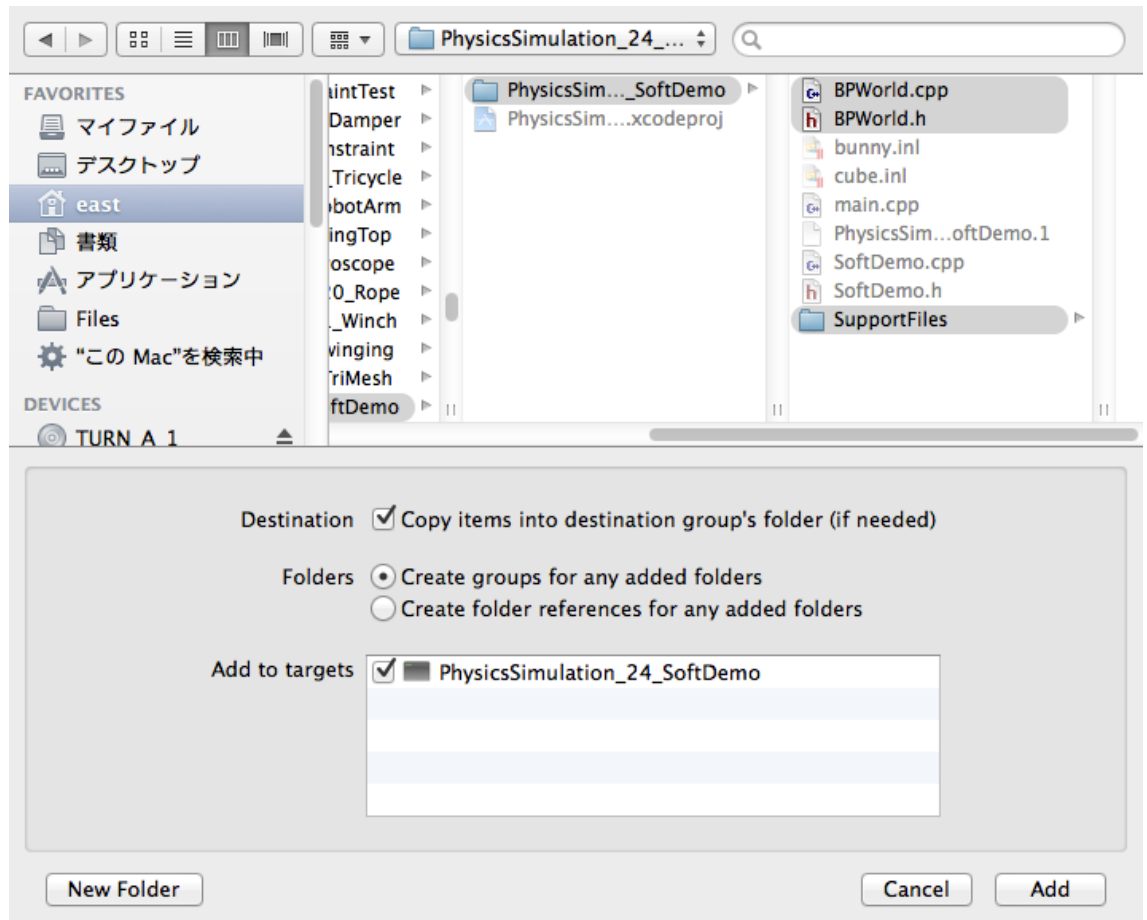
- glut.framework : GLUT の環境を提供する
- OpenGL.framework : OpenGL の環境を提供する。

→上記 2 つの framework は、「+」マークをクリックすると現れる下記のウィンドウ内の入力窓に、それぞれ「glut」、「opengl」と入力することで表示されるので、これをダブルクリックすることで Link Binary として登録される。



- **BulletCollision.framework** :  
(Bullet Physics ルートディレクトリ)/src/BulletCollision 内の拡張子が **framework** のファイルを選択する.
  - **BulletDynamics.framework** :  
(Bullet Physics ルートディレクトリ)/src/BulletDynamics 内の拡張子が **framework** のファイルを選択する.
  - **LinearMath.framework** :  
(Bullet Physics ルートディレクトリ)/src/LinearMath 内の拡張子が **framework** のファイルを選択する.  
→上記の 3 つの **framework** では、上図にあるウィンドウの下部左側の  
“Add Other ....” を選択して表示されるウィンドウ上で選択する.
- (8) Demos 内の各サンプルプログラムを動作させる際、**gcc** の **makefile** では **SUPPORTFILES** の指定で他の **cpp** ファイル、**h** ファイルを取り込むことが出来るが、**Xcode** 上ではこれは出来なさそうである。(調べると出来るかもしれない)  
その為、上記のインクルードファイル、**Library** ファイル以外でプログラムを動作させるために必要なファイルを取り込むために、「**SupprtFiles**」フォルダを作成し、これにサンプルプログラムを動作させるために必要なファイルを集めた。  
必要/不必要にかかわらず、これを取り込むことを進める.
- (9) 上記 (7) の他、下記の 2 つのファイルを取り込む.
- **BPWorld.h**
  - **BPWorld.cpp**
- (10) 上記 (7) 及び (8) のファイルを取り込む作業は、下記のように行う。  
下記のファイル及びディレクトリを、対象となるプログラムがあるディレクトリにコピー、ペーストする.
- **SupprtFiles** ディレクトリ及びその中のファイル全て
  - **BPWorld.h**
  - **BPWorld.cpp**
- (11) 下図にあるように、プロジェクトに取り込みたいファイルを選択し、**[Files]-[Add Files To ....]**を選択し、追加する。  
→ファイルは複数選択することが出来る.

- “Copy items into destination group’s folder (if needed)” にチェックを入れる．これによりオリジナルからファイルが移動してしまうことは無い．今回はコピーしたファイルを取り込んでいるのでチェックは不要だが、チェックを入れるようにしていると良い．
- この場合、BPWorld.h, BPWorld.cpp, SupportFiles ディレクトリの 3 つを選択した状態で「OK」をクリックする．



- (12) 以上の操作により，プログラムをビルドすることが可能となる．

### 3. 柔軟物のモデル作成

#### (1) frameworkの追加：

Bulletでは柔軟物のモデル化，描画も可能である．剛体に対して柔軟に変形する物体としてSoft Body (btSoftBody)の定義を使用する．

ライブラリの準備として，btSoftBodyを扱うためにインクルード，ライブラリファイルを追加する必要がある．Bulletを最低限動かすのに必要なのは，BulletCollision, BulletDynamics, LinearMathの3つであると上記2- (7) で説明したが，btSoftBodyを使う場合にはもう一つ，

BulletSoftBody.frameworkを追加する．

BulletSoftBody.frameworkの追加方法はBulletCollision.frameworkと同様である．

#### (2) インクルードファイルの追加：

インクルードファイルとライブラリファイル ソースコードを変更する前には，以前と同じようにこれまで作成したコードをフォルダごとコピーして，新しい名前のフォルダから実行するようにしておく．

まず，btSoftBodyを扱うために以下のインクルードファイルを追加する．

```
#include <BulletSoftBody/btSoftRigidDynamicsWorld.h>
#include
<BulletSoftBody/btSoftBodyRigidBodyCollisionConfiguration.h>
#include <BulletSoftBody/btSoftBodyHelpers.h>
#include <BulletSoftBody/btSoftBody.h>
```