

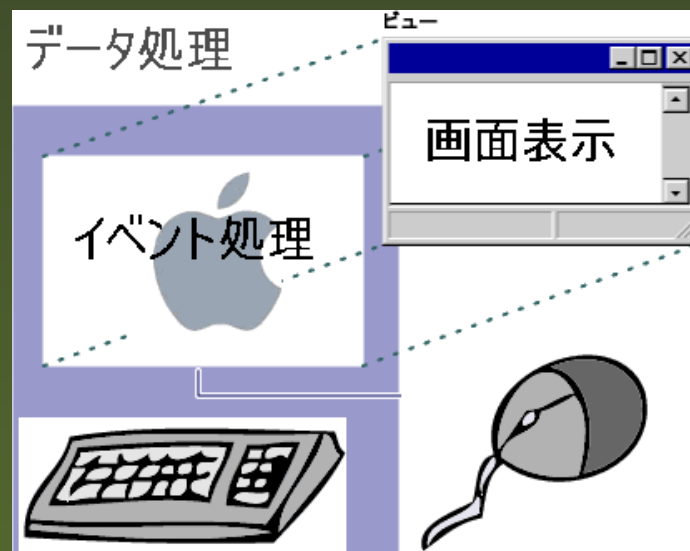
# iPhone アプリ開発の実際

石田 基広

徳島大学総合科学部

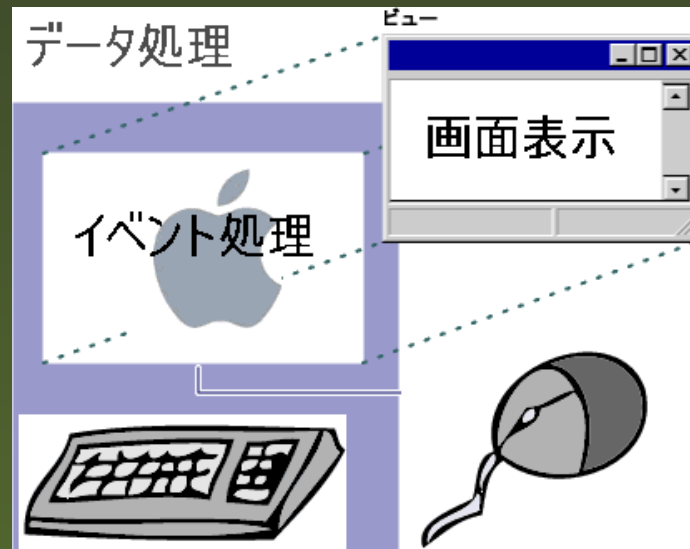
# GUI プログラミング (MVC)

- すでに提供されている部分 (Framework/API)
  - ディスプレイ画面への描画 (View)
  - マウス/キーボード操作 (イベント) の感知 (Controller)



# GUI プログラミング (MVC)

- すでに提供されている部分 (Framework/API)
  - ディスプレイ画面への描画 (View)
  - マウス/キーボード操作 (イベント) の感知 (Controller)
- ユーザーが追加する処理 (Model)
  - データの処理 (計算, ファイルの読み書き)
  - 感知された操作への反応処理



# iPhone ひな形 1

ひな形から始める App 作り

- **Nav Bar Based** (「設定」アプリ)
- **OpenGL ES** (3D ゲームアプリ)
- **Tab Bar Based** (「AppStore」アプリ)
- **Utility** (「天気」アプリ)
- **View Based** (基本ひな形)
- **Window Based** (基本の基本ひな形)「本日使う」



# iPhone ひな形 2

ひな形から始める App 作り

- Navigation-Based (「設定」アプリ)
- OpenGL ES (3D ゲームアプリ)
- **Tab Bar** (「AppStore」アプリ)
- Utility (「天気」アプリ)
- View Based (基本ひな形)
- Window Based (基本の基本ひな形)「本日使う」

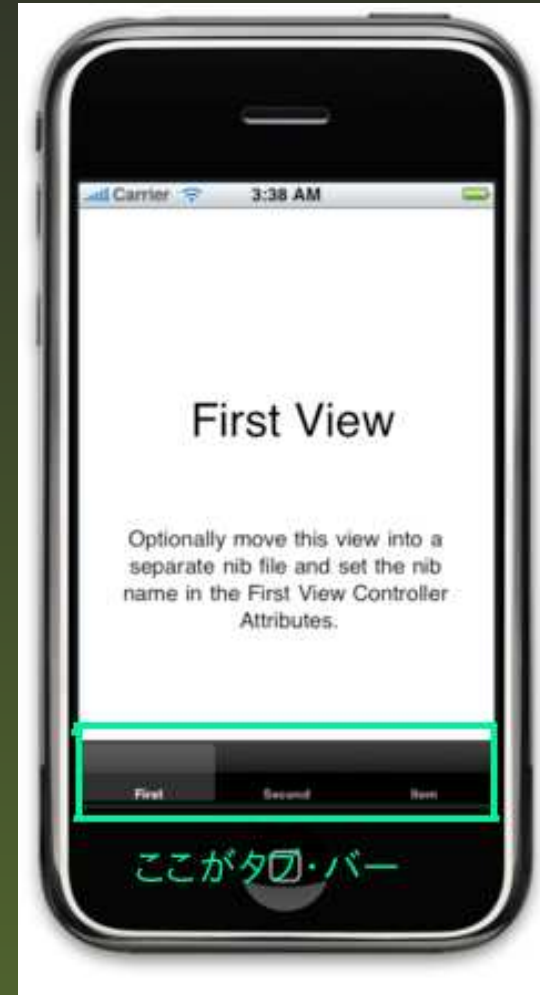


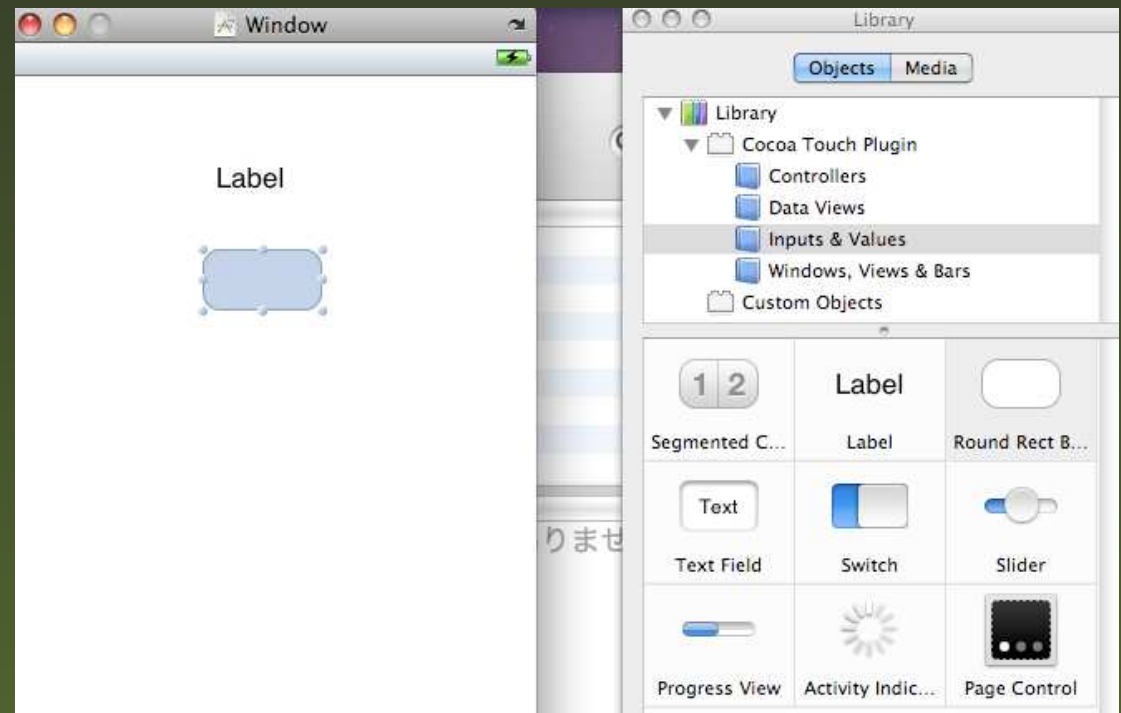
Figure 0: Tab Bar iPhone アプリ開発の実際 - p.4/14

# GUI デザイン

## Resource / MainWindow.xib

### ■ Interface Builder

- Tools ->  
Library ->  
Cocoa  
Touch
- Input &  
Values
- Label と  
Roud  
Button



# ヘッダファイルの編集

```
#import <UIKit/UIKit.h>

@interface HelloWorldAppDelegate : NSObject
<UIApplicationDelegate> {
    UIWindow *window;
    IBOutlet UILabel * label;
}

@property (nonatomic, retain) IBOutlet UIWindow *window;
-(IBAction) sayHello;

@end
```

- \*AppDelegate.h  
ヘッダ
- IBOutlet 型変数の  
定義
- -(IBAction) メソッド  
の定義

# Cソース・コードへの追記

```
(void)dealloc {
    [window release];
    [super dealloc];
}

-(IBAction)sayHello
{
    label.text = @"Hello";
}

@end
```

- \*AppDelegate.m  
ソース
- -(IBAction) メソッド  
の実装
- label.text は「ラ  
ベルという部品  
の text という  
特性」
- ○○. △△のよ  
うなドットを挟  
んだ記法はオブ  
ジェクト指向の  
特徴



# デレゲート 接続

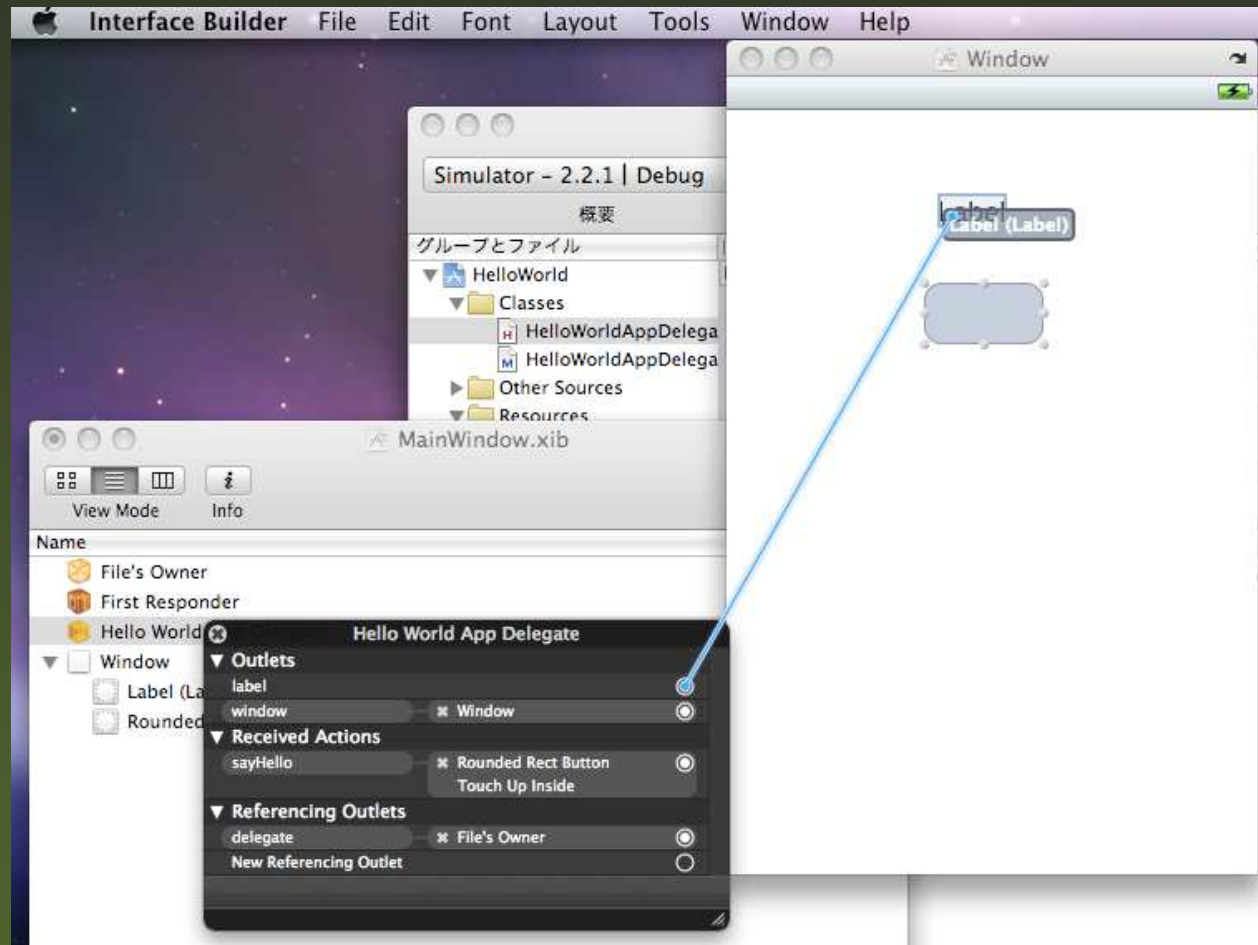
---

## 本日の難関

- MainWindow をダブルクリック
- HelloAppDelegate アイコンを右クリック
- label 変数と画像のラベルと結ぶ
- sayHello メソッドをボタンと結び
- Touch Up Inside アクションを指定

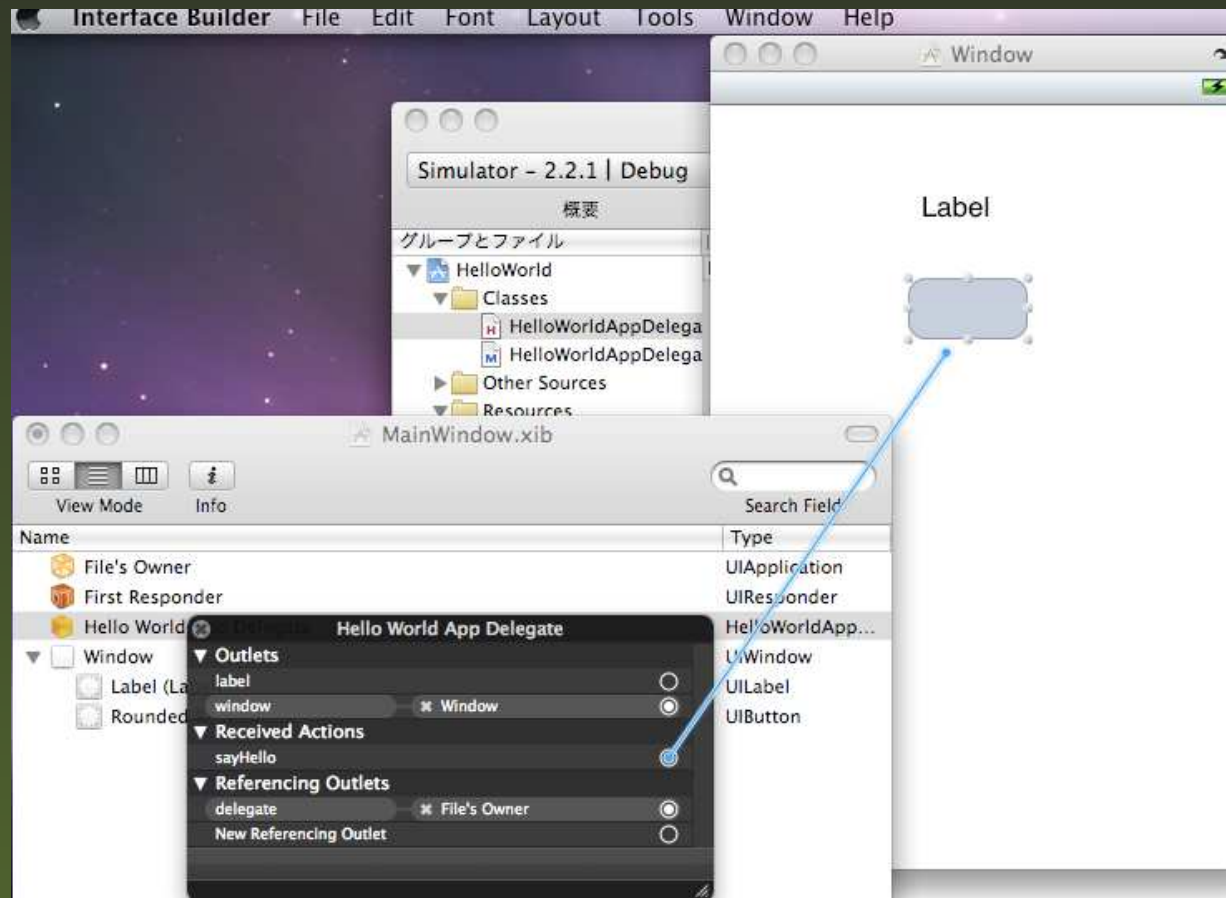
# Window Based プログラミング 2

Xcode で label 変数と GUI ラベルを接続



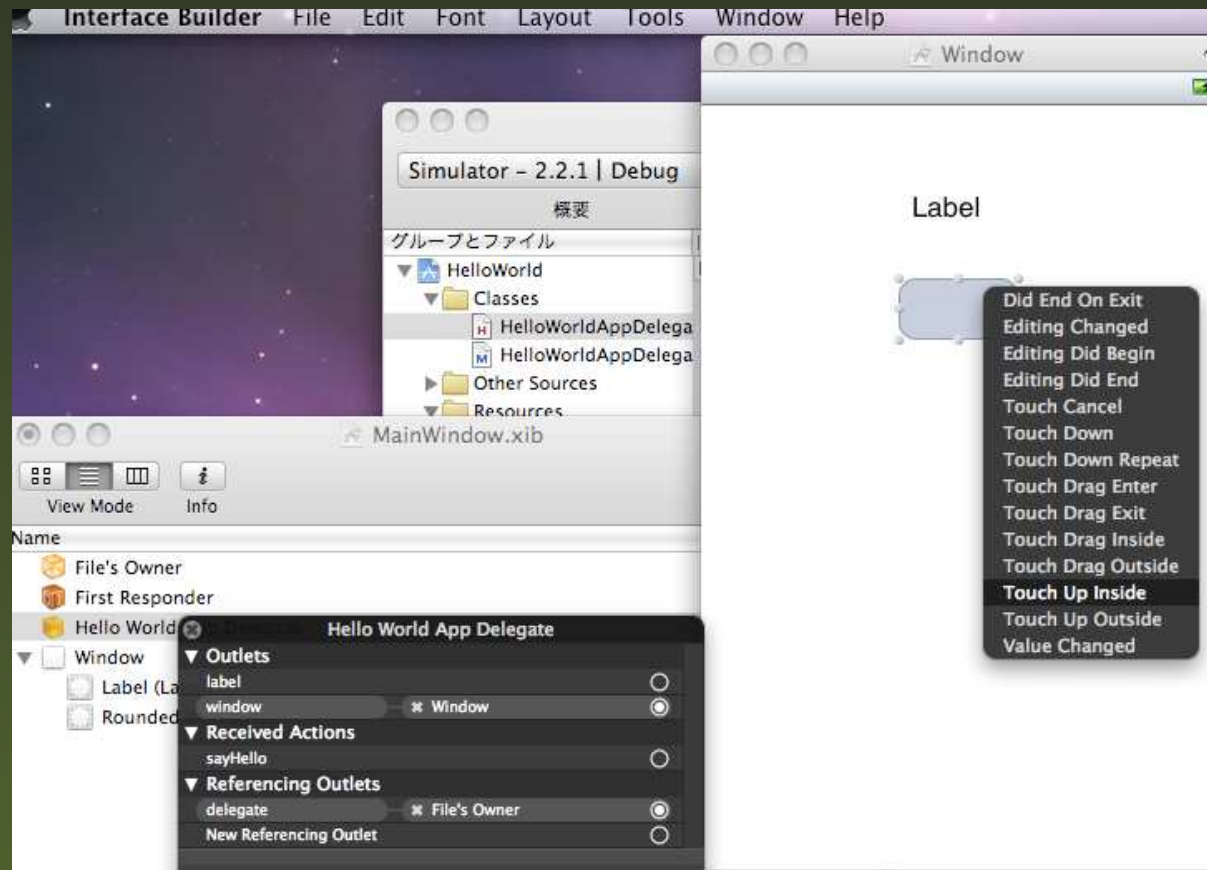
# Window Based プログラミング 3

Hello 表示メソッドと GUI ボタンを接続



# Window Based プログラミング 4

## GUI ボタンのアクションを指定



# 開発の流れ1

- Simulator でプロトタイプ作成
- Device でデバッグ
  - Provisioning の設定

The screenshot shows the Apple Developer Program Portal interface. The left sidebar contains navigation links: Home, Team, Certificates, Devices, App IDs, Provisioning (selected), and Distribution. The main content area is titled "Development Provisioning Profiles" and displays a table of profiles. The table has columns for "Provisioning Profile", "App ID", "Status", and "Actions". Each row includes a checkbox, a profile name with a "(Details)" link, an App ID, a status with an expiration date, and "Download" and "Edit" buttons.

Provisioning Profile	App ID	Status	Actions
<input type="checkbox"/> Fish (Details)	MBMUVD84L4.*	Active until 06 5 2010	Download Edit
<input type="checkbox"/> KuragePre0509 (Details)	MBMUVD84L4.*	Active until 08 5 2010	Download Edit
<input type="checkbox"/> ishida090327 (Details)	MBMUVD84L4.*	Active until 07 4 2010	Download Edit
<input type="checkbox"/> ishida20090408dev (Details)	MBMUVD84L4.*	Active until 07 4 2010	Download Edit
<input type="checkbox"/> motohiro090507 (Details)	Q566C3K9RE.*	Active until 06 5 2010	Download Edit
<input type="checkbox"/> tao (Details)	9ZHJ22UE7V.tao.*	Active until 07 5 2010	Download Edit

# 開発の流れ2

- Simulator でプロトタイプ作成
- Device でデバッグ
  - Provisioning の設定
  - ID の設定



# 開発の流れ3

- Simulator でプロトタイプ作成
- Device でデバッグ
  - Provisioning の設定
  - ID の設定
  - info.plist の設定

