



【C1】 Delphi/C++チュートリアルセッション

# 「RAD Studio XE5によるマルチ言語／ マルチデバイス開発の進め方」

エンバカデロ・テクノロジーズ  
エヴァンジェリスト 高橋智宏

# アジェンダ

- RAD Studio XE5の概要
  - Delphi or C++
  - Intel or ARM
  - Windows / OS X / iOS / Android
- プロジェクトとは?
  - FireMonkey HD アプリ
  - FireMonkey Mobile アプリ
- ユニットとは?
- データモジュールとは?
- Tips
  - 複数フォームの表示・非表示
  - Androidの[Back]キー
- まとめ



# RAD Studio XE5の概要



# Delphi XE5 / C++Builder XE5

## 統合開発環境

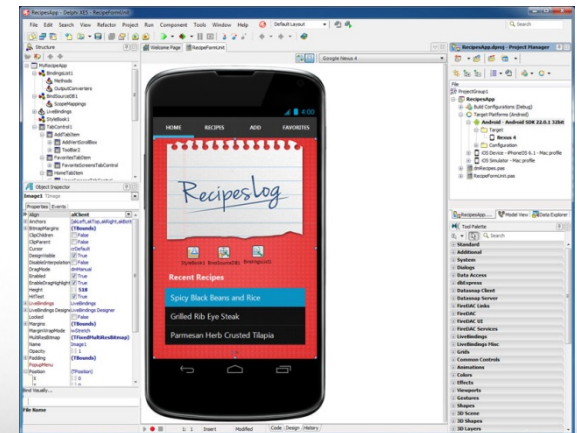
- マルチOS, マルチデバイス対応のビジュアルRADツール
  - 単一コードベースで Windows, OS X, iOS, Android に対応
- アイデアを素早くカタチに
  - 多数用意されたコンポーネントをドラッグ&ドロップして効率的にアプリを開発可能
  - もちろん、自作コンポーネントの作成&登録も可能!!
- 真のネイティブコード
  - 中間コードや仮想マシンを必要としない、CPU/GPUの性能を100%活用したハイパフォーマンスなネイティブアプリを生成

Delphi XE5 / C++Builder XE5 のマルチプラットフォーム対応状況:

	Win32	Win64	OS X	iOS	Android
Delphi	■	■	■	■	■
C++Builder	■	■	■	■	

New!

## 統合開発環境 (IDE)



Developer Camp

# なぜ、マルチデバイス&OS対応が可能なのか？

- FireMonkey(FM)フレームワーク
  - CPU / GPU ネイティブ
  - コンポーネントレイヤーがデバイスの差異を吸収
  - しかしライブラリはネイティブ



※ Androidは現在Delphi言語のみでサポート

# Windowsアプリの構造

統合開発環境(IDE)  
(エディタ、デバッガ、フォームデザイナー etc.)

Intel/AMD x86用  
Delphi/C++ コンパイラ

Intel/AMD x64用  
Delphi/C++ コンパイラ

Delphi/C++ RTL

FireMonkey

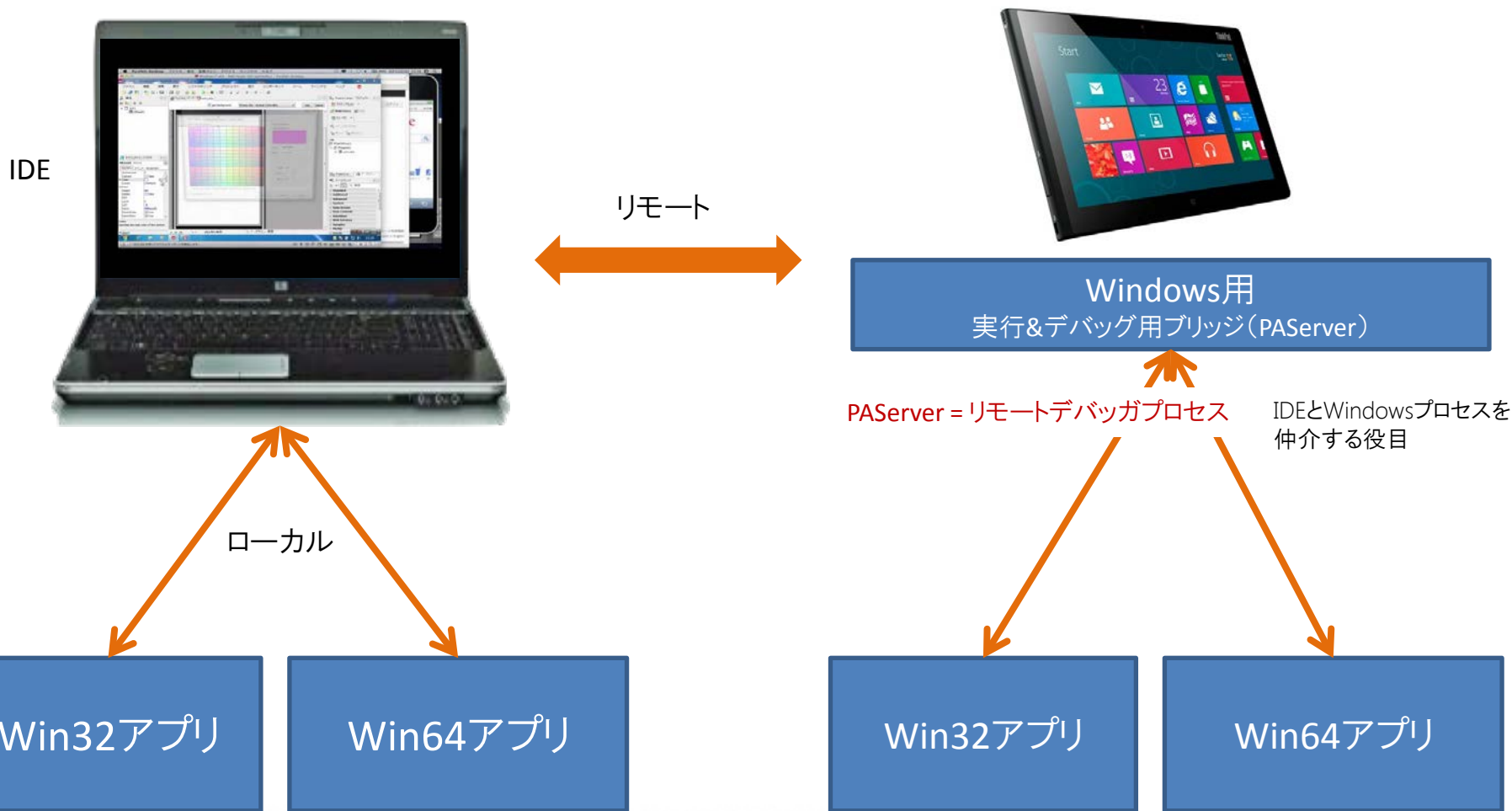
各種コンポーネント

Win32/Win64 API

DirectX

XP, Vista, 7, 8, 2008, 2012

# Windowsアプリ実行&デバッグの仕組み



# OS Xアプリの構造

統合開発環境(IDE)  
(エディタ、デバッガ、フォームデザイナー etc.)

Intel x86用  
Delphi/C++ コンパイラ

Delphi/C++ RTL

FireMonkey

各種コンポーネント

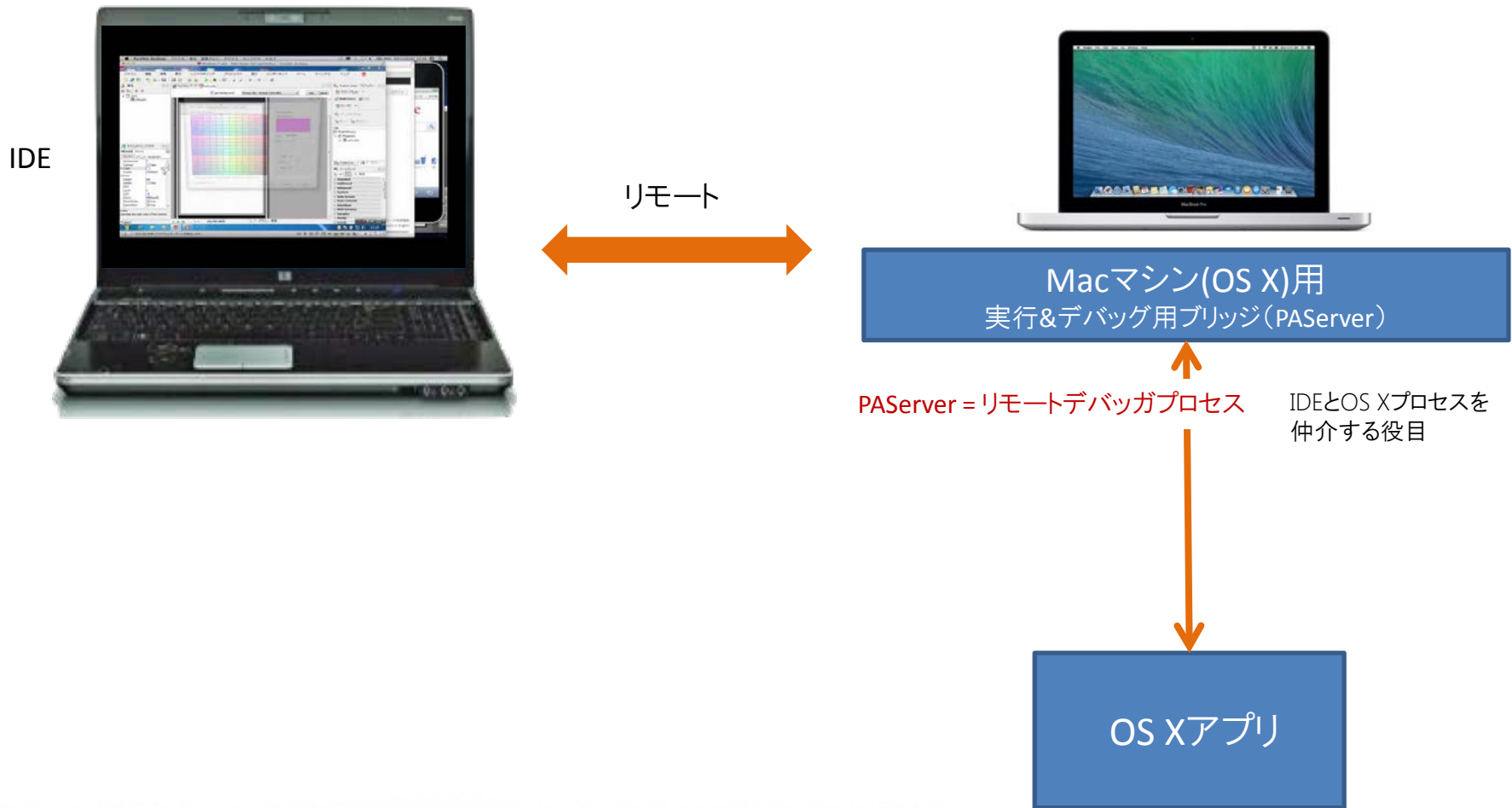
OS X API / POSIX

OpenGL

Lion, Mountain Lion, Mavericks



# OS Xアプリ実行&デバッグの仕組み



# iOSアプリの構造

Delphi 統合開発環境(IDE)  
(エディタ、デバッガ、フォームデザイナー etc.)

iOSシミュレータ(Intel x86)用  
Delphi コンパイラ

iOSデバイス(ARMv7)用  
Delphi/**C++** コンパイラ

New!

Delphi/C++ RTL

FireMonkey

各種コンポーネント

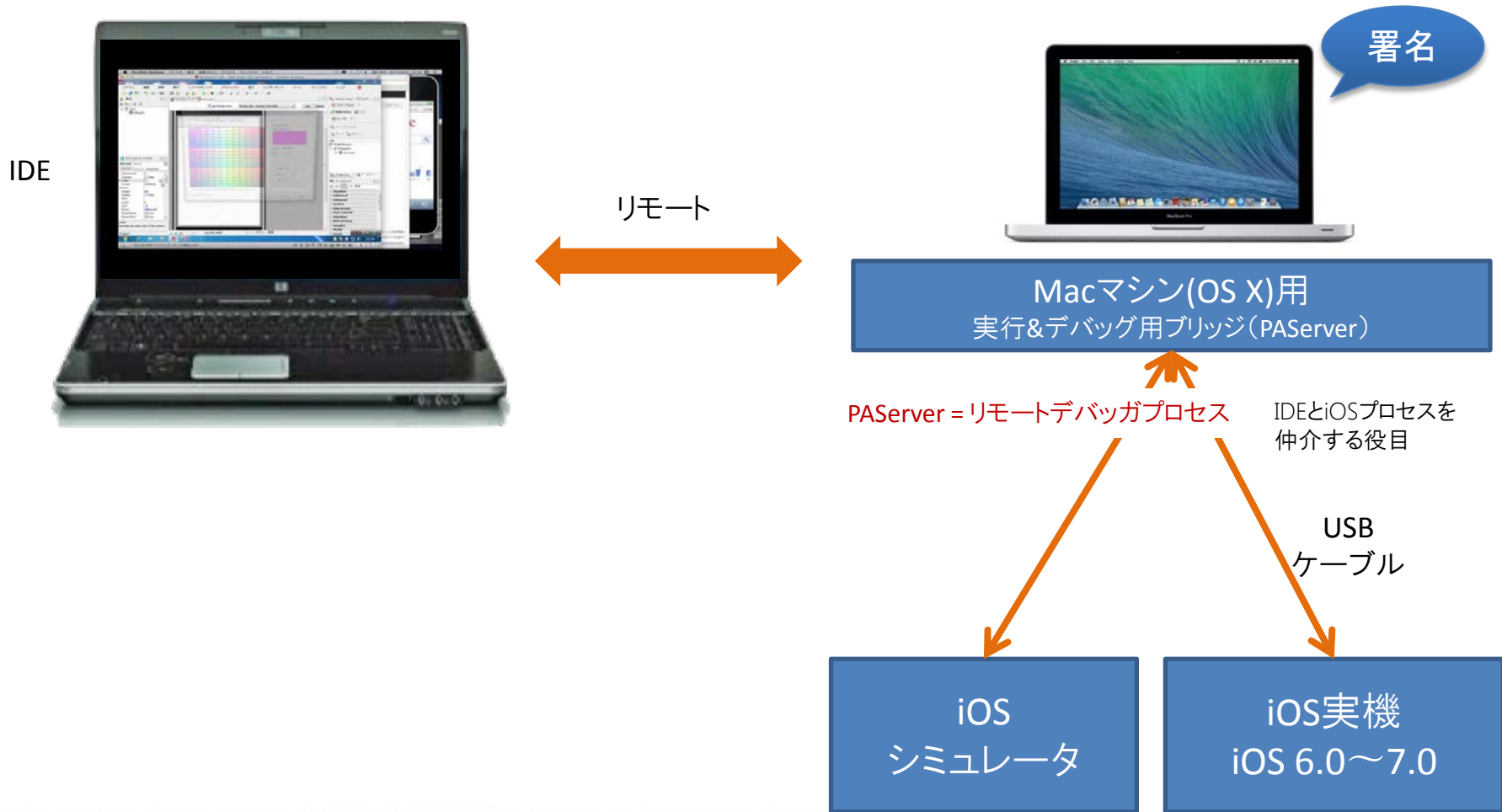
iOS API / POSIX

OpenGL ES

iOSシミュレータ(6.x,7.0)

iOSデバイス(iOS 6.x,7.0)

# iOSアプリ実行&デバッグの仕組み



# Androidの場合

Delphi 統合開発環境(IDE)  
(エディタ、デバッガ、フォームデザイナー etc.)

Android(ARMv7 + NEON)用 Delphi コンパイラ

Delphi RTL

FireMonkey

各種コンポーネント

Android SDK

Android NDK

OpenGL ES

Androidエミュレータ

Androidデバイス(2.3.3～4.4)

# Androidアプリ実行&デバッグの仕組み

署名

IDE

実機用のUSBドライバは  
適宜自分でインストール

USB  
ケーブル

実機側で、USBデバッグ  
のオプションを有効する

Android  
エミュレータ  
(ARM/GPUサポート)

Android実機  
(2.3.3～4.4)



# プロジェクトとは？

# Delphiプロジェクト

- FireMonkey HD アプリケーション

- Win32
- Win64
- OS X



ビルド構成

- Debug
- Release

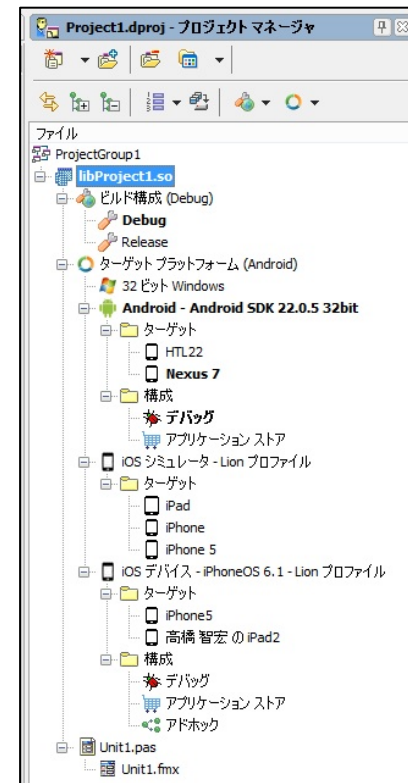
- FireMonkey モバイル アプリケーション

- iOS シミュレータ
- iOS デバイス
- Android
- Win32



ビルド構成

- Debug
- Release



プロジェクトマネージャ

# C++プロジェクト

- FireMonkey HD アプリケーション

- Win32
- Win64
- OS X



ビルド構成

- Debug
- Release

New!

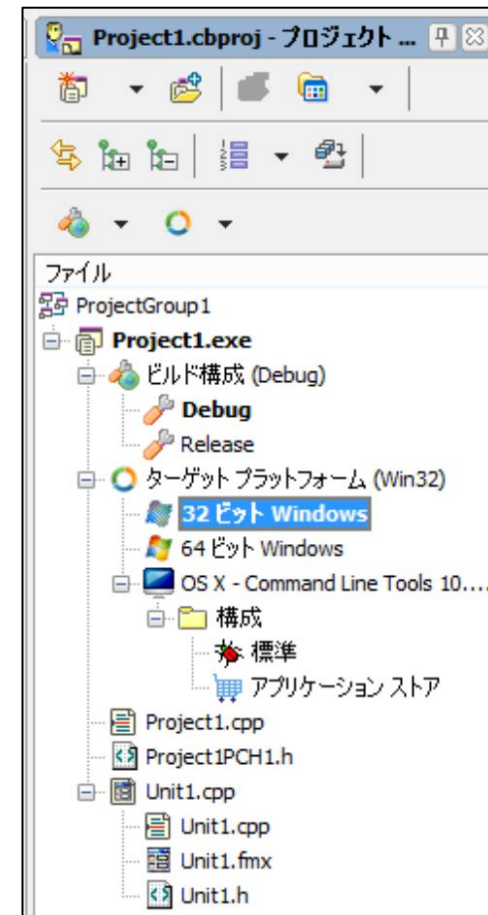
- FireMonkey モバイル アプリケーション

- iOS デバイス
- Win32



ビルド構成

- Debug
- Release



プロジェクトマネージャ



# モバイル特有のプロジェクト設定

- iOSの[バージョン情報]
  - UIDeviceFamily
    - iPhone
    - iPad
    - iPhone および iPad
- Androidの[バージョン情報]
  - package
    - デフォルトの `com.embarcadero.$(ModuleName)` を変更
- Androidの[使用する権限]
  - アプリの利用する機能により適宜有効に!!

# IDEの設定

- 自動保存の設定の[エディタファイル] を ON
- 接続プロファイルマネージャ
  - OS X : リモートでOS XアプリとiOSアプリを実行
  - 32ビットWindows : リモートでWin32アプリをDebug
  - 64ビットWindows : リモートでWin64アプリをDebug
- プロビジョニング
  - [iOSデバイス – デバッグ]に指定する証明書の文字列の意味
    - デフォルトは iPhone Developer だが...
- SDKマネージャ
  - Androidは、ローカルのWindowsマシン内のSDK/NDK/JDKを利用
- フォームデザイナーのデバイスマネージャ

# プロジェクトグループ

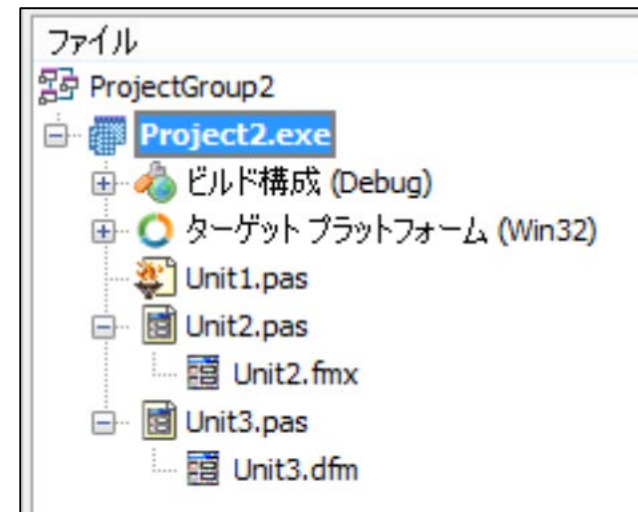
- 複数のプロジェクトをまとめるプロジェクトのこと
  - 拡張子は .groupproj
- プロジェクトのビルド順も管理
  - プロジェクトマネージャ内で上から下の順にビルド
- プロジェクト構成の例
  - ProjectGroupA
    - Delphi iOS(iPhone&iPad) & Android両用プロジェクト
    - Delphi Win32/Win64両用プロジェクト
  - ProjectGroupB
    - Delphi iOS(iPad)専用プロジェクト
    - Delphi Android専用プロジェクト



## ユニットとは?

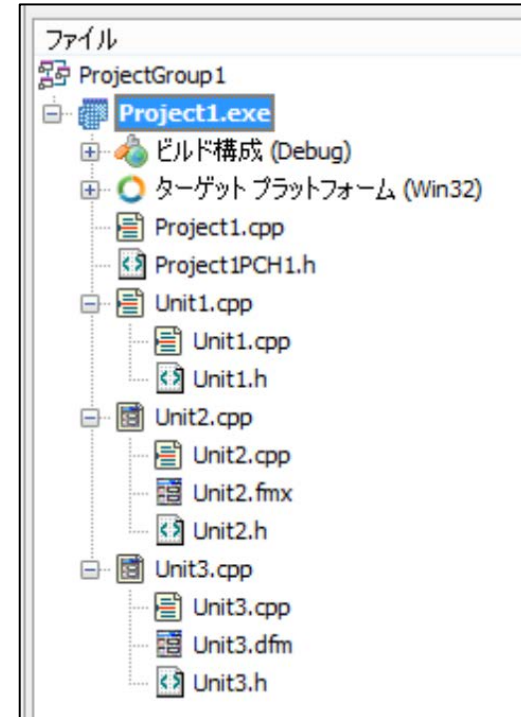
# Delphi

- プロジェクトマネージャで管理するモジュールの単位
- クラスや関数のみの場合
  - Unit1.pas
- フォーム(Unit2)の場合
  - Unit2.pas + Unit2.fmx
- データモジュール(Unit3)の場合
  - Unit3.pas + Unit3.dfm
- プロジェクトオプションの[フォーム]は要注意!!
  - メインフォーム：アプリ起動時に最初に表示されるフォーム
  - 自動生成フォーム：データモジュールも含む
  - 使用可能フォーム



# C++

- プロジェクトマネージャで管理するモジュールの単位
- クラスや関数のみの場合
  - Unit1.cpp + Unit1.h
- フォーム(Unit2)の場合
  - Unit2.cpp + Unit2.fmx + Unit2.h
- データモジュール(Unit3)の場合
  - Unit3.cpp + Unit3.dfm + Unit3.h
- プロジェクトオプションの[フォーム]は要注意!!
  - メインフォーム：アプリ起動時に最初に表示されるフォーム
  - 自動生成フォーム：データモジュールも含む
  - 使用可能フォーム



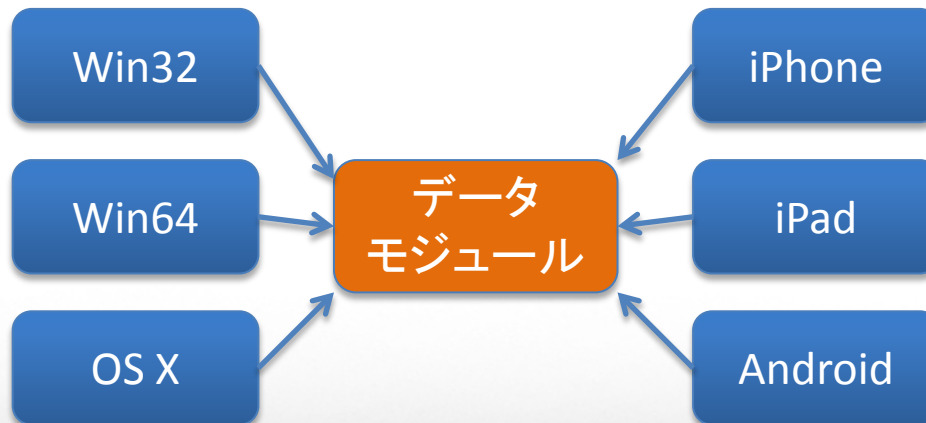


# 4

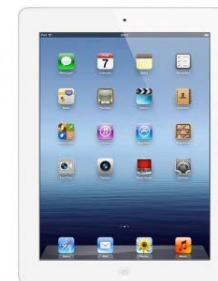
## データモジュールとは？

# データモジュール

- フォームデザイナー内で非ビジュアルコンポーネントをD&D
  - GUIコンポーネントは配置できません
- GUIに依存しない機能をカプセル化
  - クラスインターフェース等を外部に公開
- 複数プロジェクト間で共有できる
  - プラットフォーム依存のコードは条件定義などで書き分けても良い
- Delphi言語、C++言語のどちらでも使える



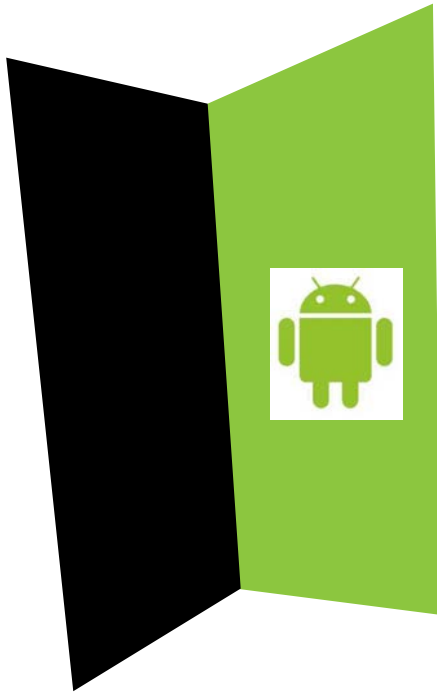




# デモ

## - 複数デバイス対応 -

# Tips



# 複数フォームの場合

- Windows / OS X
    - Show() : 子ウィンドウをモードレス表示
    - ShowModal() : 子ウィンドウをモーダル表示
  - iOS / Android
    - Show() : フルスクリーンのフォームをモードレス表示
      - 最前面に覆い被さる
    - ShowModal(無名メソッド)
      - 同じくモードレス表示だが閉じると無名メソッドが呼ばれる
- [http://blog.marcocantu.com/blog/xe5\\_anonymous\\_showmodal\\_android.html](http://blog.marcocantu.com/blog/xe5_anonymous_showmodal_android.html)

# Androidの[Back]キーへの対応

- 基本は、アプリはバックグラウンドに移動
  - キーボードが表示されている場合
  - 子フォームが表示されている場合
- 画面内のGUI更新に使えます
  - 前ページに戻る etc...
- AndroidのFormのOnKeyUpイベント
  - vkHardwareBack を監視
  - 0 を代入して、[Back]キーを自分で処理する

```
procedure TForm2.FormKeyUp(Sender: TObject; var Key: Word;  
                                var KeyChar: Char; Shift: TShiftState);  
begin  
  [$IFDEF ANDROID]  
    if Key = vkHardwareBack then  
      begin  
        // ...  
        Key := 0; // 0代入で[Back]キーを自前でハンドリング  
      end;  
    [$ENDIF]  
  end;
```