

Localized Your App Tutorial

Author by Glee

Table

前言	1
增加語言	1
編輯 xliff 檔案	2
匯入 xliff 檔案	3
在程式碼中呼叫翻譯檔	3
產生 Localizable.strings 語系檔	4
自動產生 Localizable.strings 語系檔	4
編輯 Localizable.strings 語系檔	5
合併 Localizable.strings 語系檔	5
Apple 參考資料	6

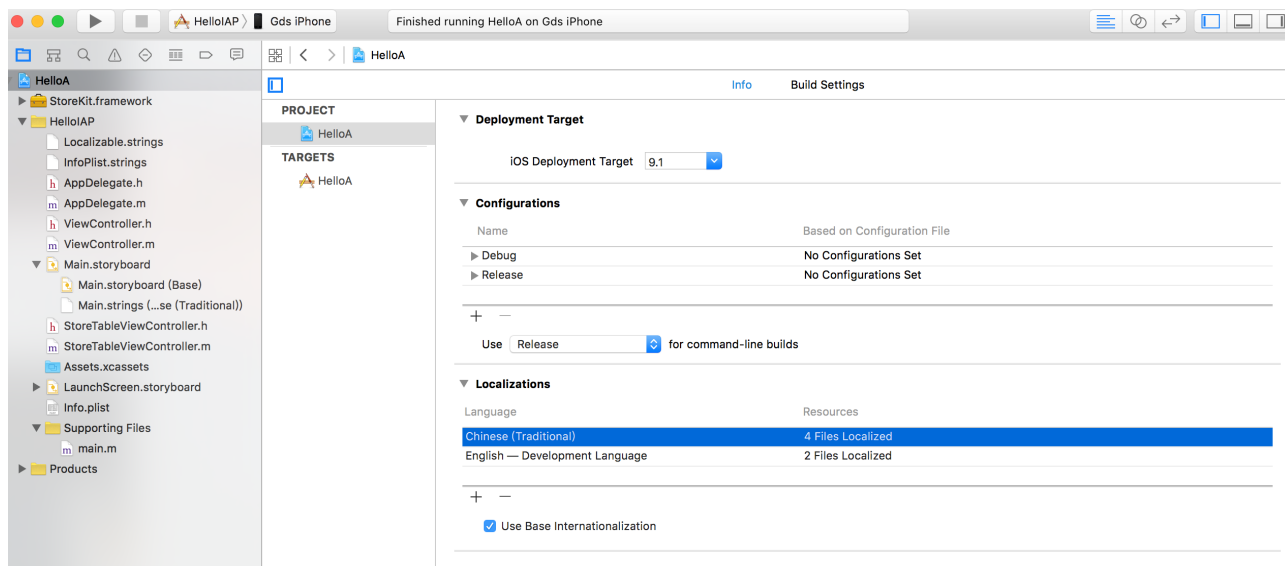
前言

將你的 App 翻譯至全球數百種 Apple iOS 支援的語言，不僅可以讓當地使用者更輕易的操作你的 App，甚至能提升不同地區的下載率。這些都要感謝 Apple 的裝置，包含 iOS 設備、Mac 設備以及 Apple Watch、Apple TV 等，全部都是支援國際化的作業系統，只要使用者在設定裡更改系統語言，你的 App 就也會跟著切換為當地語言，只要你有為該語系翻譯的話。

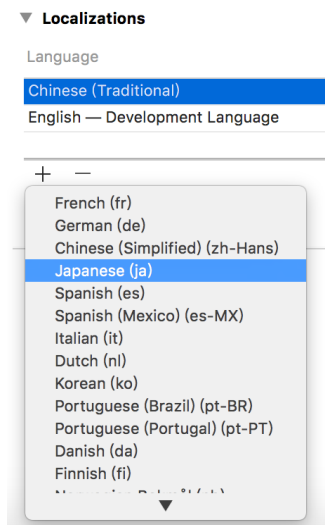
在 App 裡提供本土化是一件非常容易的事，你可以將 Storyboard、Xib、String 檔等匯出成容易閱讀的文字檔或 XML Localisation Interchange File Format（簡稱 xliiff 檔），這些連一般文字編輯器都能開啟的檔案，再外包給專業的翻譯社，就能完成翻譯了。

增加語言

預設專案開啟時，會使用英文做為基礎語言（base localization），當要新增一個語系時，需要至專案裡增加。



在 Localizations 裡按一下「+」，新增其它語系



將需要翻譯的檔案都選起來，匯出成 Localizable Strings 檔案。

<input checked="" type="checkbox"/>	Main.storyboard	Base	⇅ Localizable Strings ⇅
<input checked="" type="checkbox"/>	LaunchScreen.storyboard	Base	⇅ Localizable Strings ⇅

接著，從 Editor > Export for localization，將專案內的檔案輸出成 xliiff 檔案，這其中包含了 Storyboard 的翻譯檔，以及其它的字串檔案（String File）。

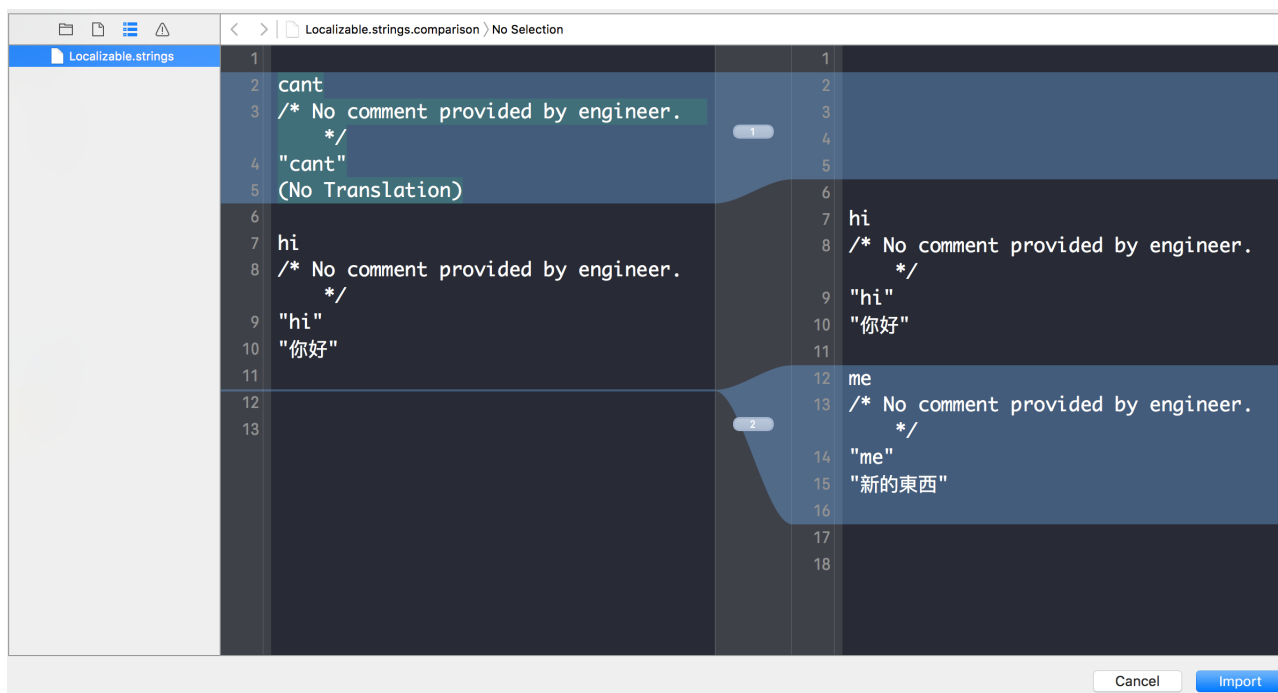
編輯 xliiff 檔案

匯出後的名稱會是語系的識別名，例如 zh-Hant.xliiff（繁體中文語系），這個檔案可以用一般的文字編輯器開啟。你只需要把翻譯結果寫在 Target 的位置就可以了。

```
<trans-unit id="J3e-tw-qdZ.normalTitle">
  <source>Restore</source>
  <target>回復</target> //翻譯結果填在這裡
  <note>Class = "UIButton"; normalTitle = "Restore"; ObjectID = "J3e-tw-qdZ";</note>
</trans-unit>
```

匯入 xliff 檔案

完成後的 xliff 檔案，可以用匯入的方式重新進入專案。透過這種方式，能保留原本已經翻譯好的內容，只匯入新的翻譯內容。請從 Editor > Import Localizations... 選擇正確的 xliff 檔案。



在程式碼中呼叫翻譯檔

iOS App 的翻譯，也可以透過查表的方式完成。例如有一個 UIAlertController，在程式碼的撰寫過程中，會帶入一些文字訊息，像是：

```
UIAlertController * controller = [UIAlertController alertControllerWithTitle:@"抬頭" message:@"訊息內容" preferredStyle:UIAlertControllerStyleAlert];
```

其中像是「抬頭」與「訊息內容」都要翻譯成不同語系時，就可以藉助查表的力量，變成：

```
UIAlertController * controller = [UIAlertController alertControllerWithTitle:NSLocalizedString(@"aTitle", nil) message:NSLocalizedString(@"aMessage", nil) preferredStyle:UIAlertControllerStyleAlert];
```

NSLocalizedString()是一個查表函數，它會去查語系字串檔中，Key 為 aTitle 及 aMessage 的文字內容，而這個字串檔將由你自行產生。

產生 Localizable.strings 語系檔

Localizable.strings 是一個語系檔，若一個專案中支援三種語系，就會有三個 Localizable.strings 檔案，分別放在語系資料夾中。

這個檔案你可以手動產生，從 File > New File > Resource > Strings File 建立，名稱務必是 Localizable.strings（大小寫一致），並放在語系資料夾中。如果要建立第二個 Localizable.strings 檔案時，只要從側邊欄中勾選語系即可。



自動產生 Localizable.strings 語系檔

另外一種方法更為簡便，無需事先產生 Localizable.strings 語系檔，只需要在撰寫程式的過程中使用 NSLocalizedString() 函式，就能用終端機的 genstrings 指令，自動在程式碼裡尋找所有需要翻譯的文字，並生成為 Localizable.strings。

genstrings 程式碼的.m檔位置 -o 要輸出的位置

或是請 genstrings 自動搜尋所有 *.m 的程式碼檔案，並輸出成 Localizable.strings，放到 zh-Hant 的資料夾下。首先，請確認終端機已經切換到專案的目錄底下。

genstrings -o zh-Hant.lproj *.m

編輯 Localizable.strings 語系檔

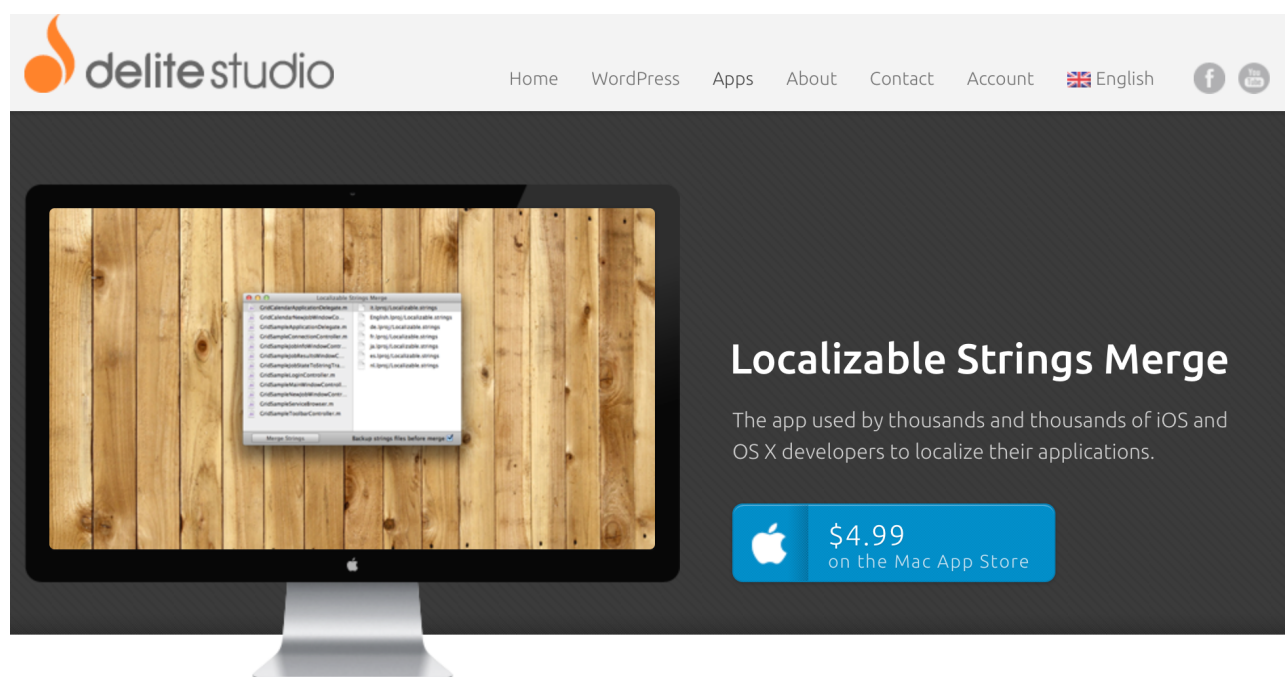
在 Localizable.strings 檔案裡，每一行都代表一對翻譯，結尾要用分號區隔。寫法很容易，等號前方是 Key，等號後方是翻譯的結果。而你也可以在這個檔案裡使用 `/**/` 備註語法。

```
/* No comment provided by engineer. */  
"hi" = "你好";  
/* No comment provided by engineer. */  
"me" = "新的東西";
```

合併 Localizable.strings 語系檔

要新增新的翻譯時，可以再匯出一次 xliff 檔。在 xliff 檔中編輯新的翻譯即可。或是使用 Glee 推薦的軟體（需付費）來合併語系檔

<http://www.delitestudio.com/app/localizable-strings-merge/>



The screenshot shows the homepage of delitestudio.com. The navigation bar includes links for Home, WordPress, Apps, About, Contact, Account, and a language selector set to English. The main content area features a large image of a computer monitor displaying the 'Localizable Strings Merge' application interface. To the right of the monitor, the text reads 'Localizable Strings Merge' followed by 'The app used by thousands and thousands of iOS and OS X developers to localize their applications.' Below this, a blue button with the Apple logo indicates the app is available for \$4.99 on the Mac App Store.

Apple 參考資料

1. Apple Internationalization and Localization Guide

<https://goo.gl/WCAsrV>