本文同步發至個人知識庫<http://www.evernote.com/l/ABDSo4vlyXRJt5No0ywNBBTBHVv8Kl_J3EM/>

|  |
| --- |
| 參考資料 |
| Delphi Android 将Google ZXing 整合 (XE5) <http://blog.csdn.net/flcop/article/details/20240703> Launching activities and handling results in Delphi XE6 Android apps <http://blong.com/Articles/DelphiXE6AndroidActivityResult/ActivityResult.htm#BarCodeScanActivity>  Delphi XE6のAndroidアプリケーションで、Intentを使ってアクティビティを開始して戻り値を取得するサンプル <http://www.gesource.jp/weblog/?p=6557> |

【注意】

1. 本範例因為不需要在 Delphi 內建立 Java 的 Object，所以不用 Java2OP.exe

2. 如果最後打包出的dex不對，可以用 d2j-dex2jar.bat classes.dex還原成 jar 檔確認檔案內容對不對

指令如下

|  |
| --- |
| d2j-dex2jar.bat classes.dex |

3.

下面的【一】和【二】只是為了要把ZXing的 Source 打包為一個  CreateActivity.jar 檔案，過程較為冗長。

如果想節省時間，可以到  <http://pan.baidu.com/s/1bn8KUcf> 下載已經打包好的 JAR 檔案，版本是Zxing 2.3 版。

當然如果想試前兩個部份，請先準備好 Eclipse 和 Java 工具來產生 CreateActivity.jar。

本步驟分成三大部份：

【一】取得Zxing的 Android Source Code

【二】使用 Android Java 產生 jar 檔並改Zxing原始碼，然後產生程式

【三】使用 Delphi XE7 加入JavaLibrary後，呼叫Zxing相機

【一】取得Zxing的 Android Source Code

原始碼下載

[https://code.google.com/p/zxing/downloads/detail?name=ZXing-2.3.0.zip&can=2&q=](https://code.google.com/p/zxing/downloads/detail?name=ZXing-2.3.0.zip&amp;amp;can=2&amp;amp;q=)

解壓後如下圖

|  |
| --- |
| 重點是三個資料夾  (1)core：核心 Library  (2)android-integration： android 整核 core 會用到的 Library  (3)android： Zxing 掃描工具   C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(31).png |

【二】使用 Android Java 產生 jar 檔並改Zxing原始碼，然後產生程式

開啟 Eclipse，建立一個專案

|  |
| --- |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(79).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(80).png |

把 core 和 android-integration 打包成 jar

|  |
| --- |
| (一) core 的部份  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(81).png   C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(82).png   C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(83).png   C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(84).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(85).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(86).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(87).png   (二) android-integration 的部份 C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(88).png       C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(89).png      C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(90).png |

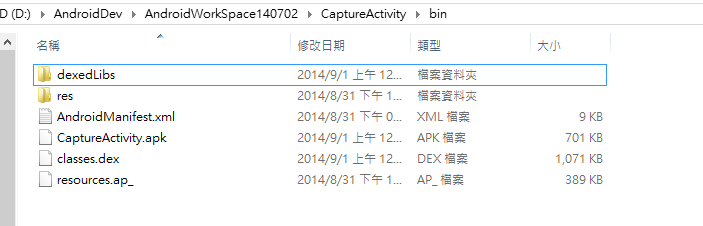
編譯Zxing拍照部份

|  |
| --- |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(91).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(92).png   在專案目錄下建立 libs 資料夾，並將兩個 jar 檔複製過來 C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(93).png   剛開始專案會有錯誤，這是因為還沒加入剛剛打包好的兩個 jar 檔。 專案右鍵=> properties C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(94).png   C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(95).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(96).png     這時程式應該不會有錯誤 C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(97).png  这个地方说明一下，红框应标在文件夹[libs]  这个地方说明一下，直接点Add JARs选择不到打包好的.jar文件，只能点击Add External JARs…才能加入。  然后编译过后，可以删除之前的，用Add JARs加入。  Build 看看，沒問題就 OK 了  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(98).png |

修改程式碼，以便讓 Delphi 進行CallBack

|  |
| --- |
| 注釋掉 onCreateOptionsMenu 事件 C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(99).png  先 Find 找到 onResume 事件 C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(100).png  注釋掉下面兩塊程式碼 C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(101).png   C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(102).png  完成後再一次 Build 專案 |

編譯後到專案目錄找classes.dex

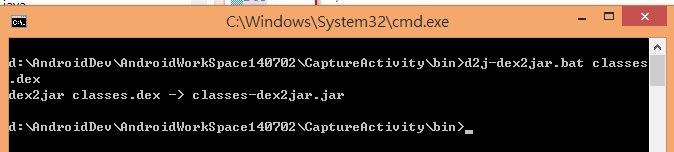


这个地方说明一下：须下载将C:\Users\Administrator\AppData\Roaming\Tencent\Users\1601681656\QQ\WinTemp\RichOle\0AJUS59DT}3L6T{MR[3`X48.png

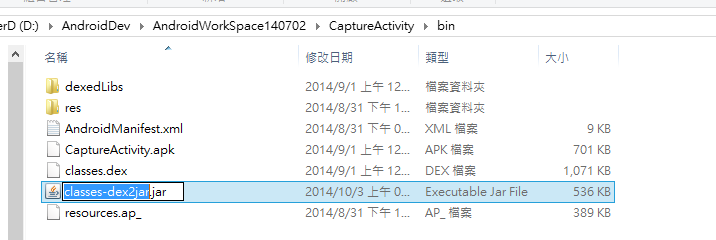
全部文件拷贝到\bin目录下，然后执行

 d2j-dex2jar.bat classes.dex

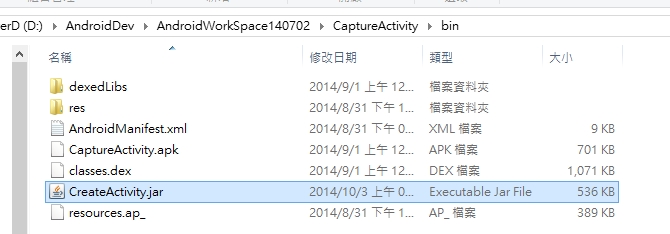
到 cmd.exe 執行 d2j-dex2jar.bat classes.dex取得 Jar 檔案



把解出的 .jar 檔案更名

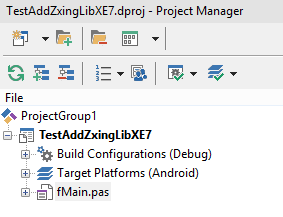


 CreateActivity.jar 取得



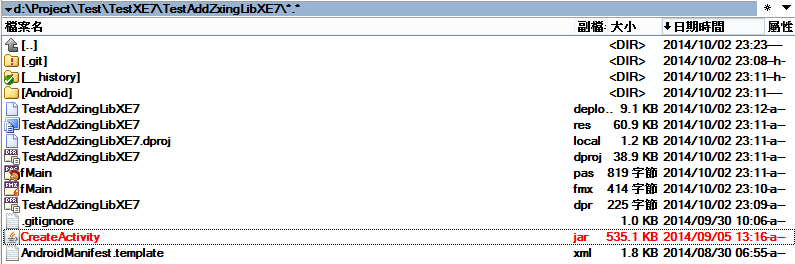
【三】使用 Delphi XE7 加入JavaLibrary後，呼叫Zxing相機

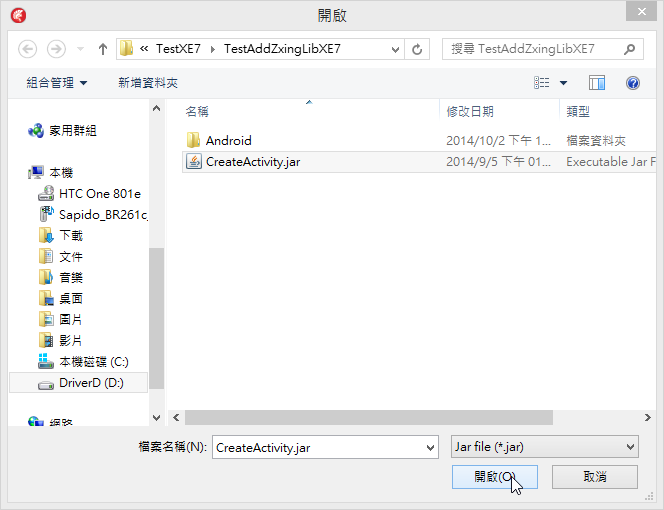
新建立一個 Delphi XE7 的專案

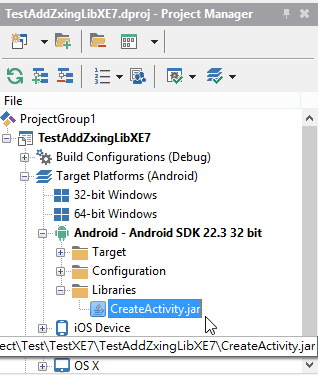
  
  
依照Zxing會用到的 Android 權限，在 project > options > uses permission 加上權限

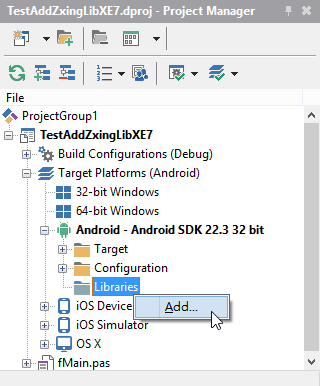
|  |
| --- |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(34).png |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(110).png |

把解好的 Jar 檔案放到專案目錄下





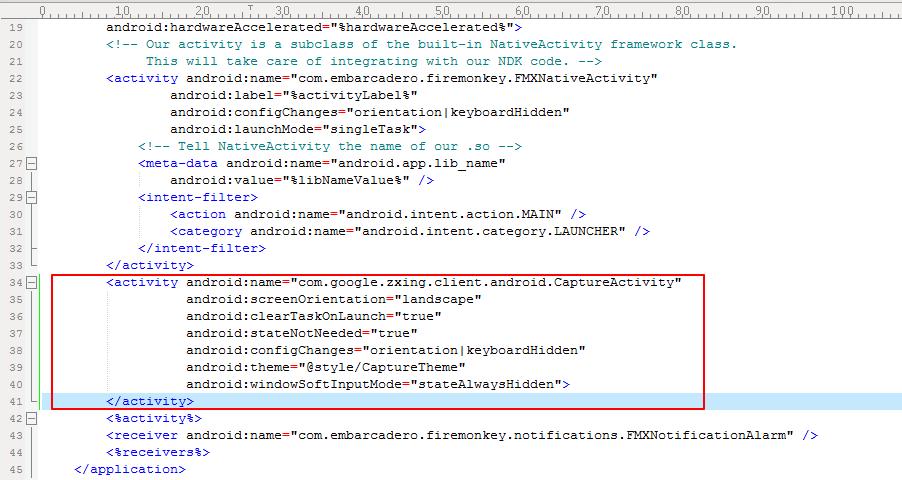




然後Compile編譯一次，產生 AndroidManifest.template.xml ，但不要 Run 手機的 App

|  |
| --- |
| 【注意】  如果只加了 Lib ，然沒有改 AndroidManifest.xml ， App 在呼叫 BarCode 時會 ANR 沒反應。  有開 DDMS 的話會看到下面這個訊息 |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(114).png |

修改AndroidManifest.template.xml ，加入CaptureActivity的區段



這個時候編譯會出現下面錯誤訊息，表示找不到 Android 的資源

|  |
| --- |
| [PAClient Error] Error: E2312 D:\Project\Test\TestXE7\TestAddZxingLibXE7\Android\Debug\TestAddZxingLibXE7\AndroidManifest.xml:44: error: Error: No resource found that matches the given name (at 'theme' with value '@style/CaptureTheme').  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(116).png |

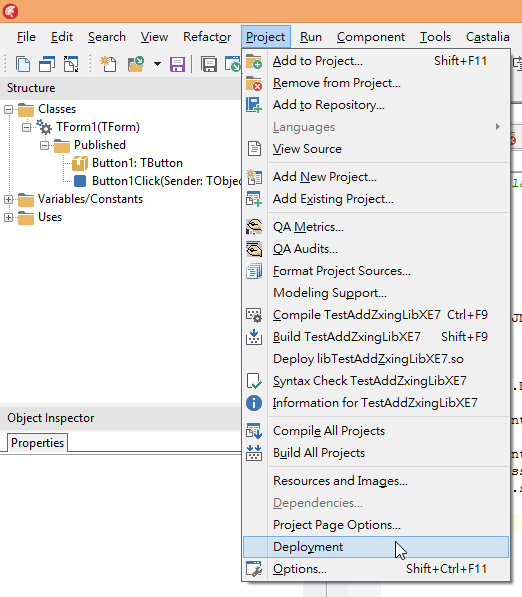
接下來要合併 Android 和 Embarcadero 的資料檔案

这个地方说明一下：注意是编译后的CaptureActivity\res文件夹

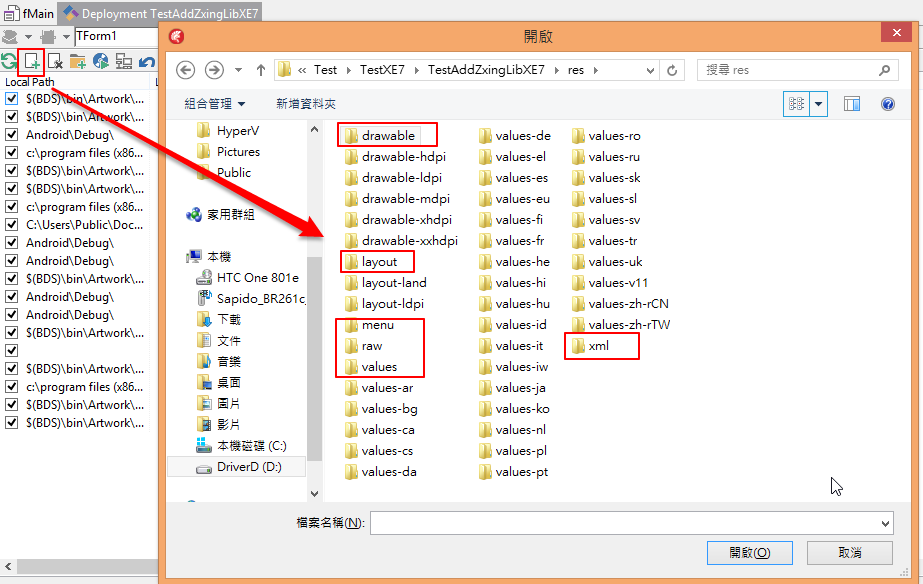
把 Android 的資源複製到專案目錄

|  |
| --- |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(117).png  C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(118).png |

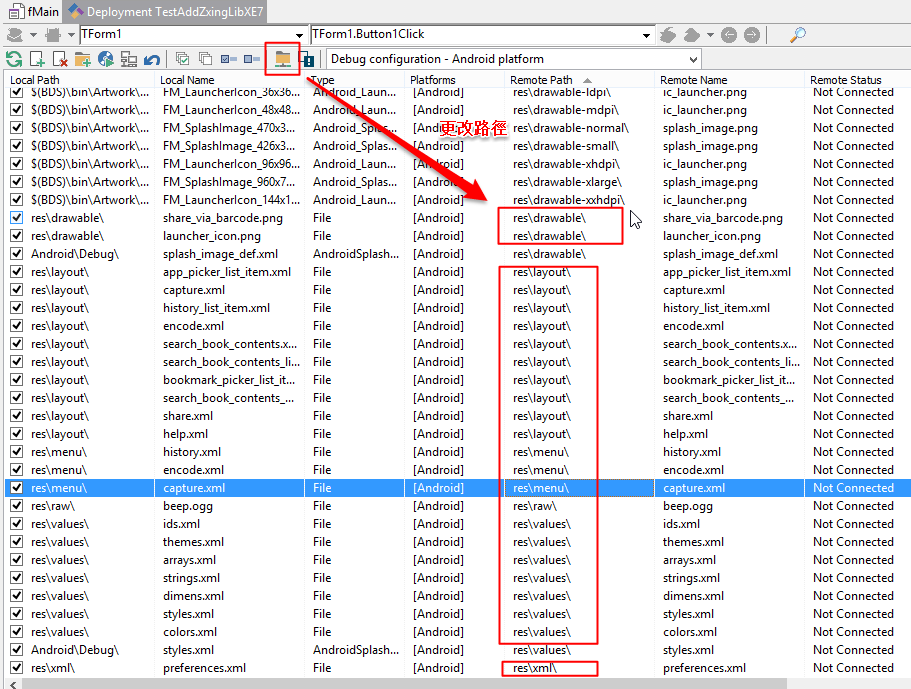
打開Deployement功能



把drawable, layout, menu, raw, values 和 xml 目錄下的所有檔案加入

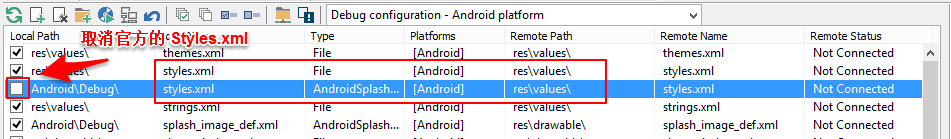


記得更改手機佈置後的路徑 (Remote Path)

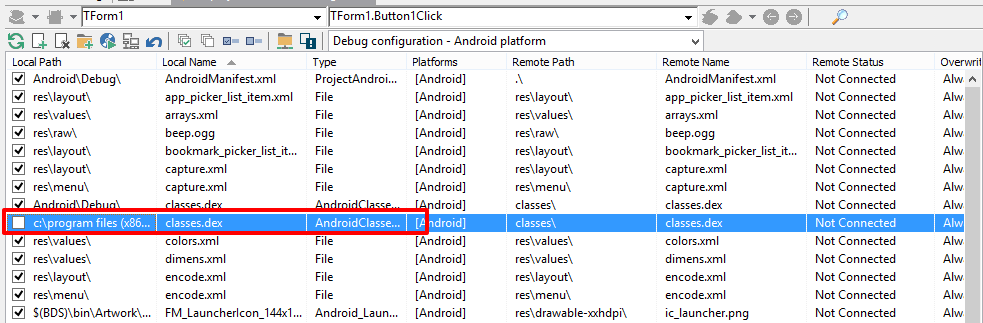


接下來因為Zxing和 Embarcadero 都有相同的檔案 res\values\styles.xml。

所以要取消其中一個，並手工把兩個檔案的內容合併



 如果有官方的classes.dex，務必取消。



这个地方说明一下：开始可能没有官方的classes.dex，

但如果发现编译出错后，请再检查一下。

把 (專案目錄)\Android\Debug\styles.xml 和 (專案目錄)\res\values\styles.xml 內容合併後如下

|  |
| --- |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(122).png  这个地方说明一下：如果加了启动界面的，也必须合并styles.xml，将开机画面的styles.xml语句，加到res\values\styles.xml文件最前面位置。 |
| <?xml version="1.0" encoding="UTF-8"?> <!-- Copyright (C) 2012 ZXing authors  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at        http://www.apache.org/licenses/LICENSE-2.0  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. --> <resources>    <style name="ResultButton" parent="@android:style/Widget.Holo.Button.Borderless.Small">     <item name="android:layout\_width">0dip</item>     <item name="android:layout\_height">wrap\_content</item>     <item name="android:layout\_weight">1</item>   </style>   <style name="ShareButton" parent="@android:style/Widget.Holo.Button.Borderless.Small">     <item name="android:layout\_width">wrap\_content</item>     <item name="android:layout\_height">wrap\_content</item>   </style> <style name="AppTheme" parent="@android:style/Theme.NoTitleBar"> <item name="android:windowBackground">@drawable/splash\_image\_def</item> <item name="android:windowNoTitle">true</item> </style> </resources> |

最後加上程式碼就可以掃瞄條碼了

|  |
| --- |
| procedure TForm1.Button1Click(Sender: TObject); var   LIntent: JIntent; begin   LIntent := TJIntent.JavaClass.init;   LIntent.setClassName(SharedActivityContext, StringToJString('com.google.zxing.client.android.CaptureActivity'));   SharedActivity.startActivityForResult(LIntent, 0); end; |

如果需要取得結果，則需要用到 TMessageManager類別。System.Messaging

这个地方说明一下：TMessageManager

须加System.Messaging单元

|  |
| --- |
| C:\Users\Cuteman\AppData\Local\Temp\enhtmlclip\Image(78).png |
| 程式碼 |
| type   TForm1 = class(TForm)     Button1: TButton;     edtScanFormat: TEdit;     edtScanContent: TEdit;     Label1: TLabel;     procedure Button1Click(Sender: TObject);   private     { Private declarations }     constScanRequestCode = 0;     varFMessageSubscriptionID: Integer;     function OnActivityResult(RequestCode, ResultCode: Integer; Data: JIntent): Boolean;     procedure HandleActivityMessage(const Sender: TObject; const M: TMessage);   public     { Public declarations }   end;  var   Form1: TForm1;  implementation  uses Androidapi.Helpers, Androidapi.JNI.App;  {$R \*.fmx}  procedure LaunchQRScanner(RequestCode: Integer); var   Intent: JIntent; begin   Intent := TJIntent.JavaClass.init;   Intent.setClassName(SharedActivityContext, StringToJString('com.google.zxing.client.android.CaptureActivity'));   //如果要預定掃瞄格式，UnComment 下面文字   //Intent.putExtra(StringToJString('SCAN\_MODE'), StringToJString('QR\_CODE\_MODE'));   SharedActivity.startActivityForResult(Intent, RequestCode); end; procedure TForm1.Button1Click(Sender: TObject); var   Intent: JIntent; begin   FMessageSubscriptionID := TMessageManager.DefaultManager.SubscribeToMessage(TMessageResultNotification,     HandleActivityMessage);   LaunchQRScanner(ScanRequestCode); end;  procedure TForm1.HandleActivityMessage(const Sender: TObject;   const M: TMessage); begin   if M is TMessageResultNotification then     OnActivityResult(TMessageResultNotification(M).RequestCode, TMessageResultNotification(M).ResultCode,       TMessageResultNotification(M).Value); end;  function TForm1.OnActivityResult(RequestCode, ResultCode: Integer;   Data: JIntent): Boolean; var   ScanContent, ScanFormat: string; begin   Result := False;    TMessageManager.DefaultManager.Unsubscribe(TMessageResultNotification, FMessageSubscriptionID);   FMessageSubscriptionID := 0;    if RequestCode = ScanRequestCode then   begin     if ResultCode = TJActivity.JavaClass.RESULT\_OK then     begin       if Assigned(Data) then       begin         ScanContent := JStringToString(Data.getStringExtra(StringToJString('SCAN\_RESULT')));         ScanFormat := JStringToString(Data.getStringExtra(StringToJString('SCAN\_RESULT\_FORMAT')));         edtScanFormat.Text:= ScanFormat;         edtScanContent.Text:= ScanContent;         Label1.Text:= '掃瞄成功';       end;     end     else if ResultCode = TJActivity.JavaClass.RESULT\_CANCELED then     begin       Label1.Text:= '掃瞄取消';     end;     Result := True;   end; end; |

最後可以去 <http://www.barcode-generator.org/> 驗證 Barcode 是否可掃

关于扫描时竖屏：

但是android提供的SDK（android.hardware.Camera）里大概不能正常的使用竖屏（portrait layout）加载照相机，当用竖屏模式加载照相机时会产生以下情况：1. 照相机成像左倾90度（倾斜）；2. 照相机成像长宽比例不对（失比）。

基本上解决办法如下：

1、在AndroidManifest.xml里面配置一下 ，使CaptureActivity属性为portrait： android:screenOrientation="portrait"

2、如果只是单纯的想改变照相机成像的方向，只需要在包com.google.zxing.client.android.camera下的 CameraConfigurationManager类中增加方法

protected void setDisplayOrientation(Camera camera, int angle) {

Method downPolymorphic;

try {

downPolymorphic = camera.getClass().getMethod(

"setDisplayOrientation", new Class[] { int.class });

if (downPolymorphic != null)

downPolymorphic.invoke(camera, new Object[] { angle });

} catch (Exception e1) {

}

}

然后在方法void setDesiredCameraParameters(Camera camera)｛｝中调用，

setDisplayOrientation(camera, 90);

具体位置在camera.setParameters(parameters);语句前面。

2014-10-15

楼主上面增加过程的方法在我的(ZXing2.3)CameraConfigurationManager中出错。

实际只需setDisplayOrientation(camera, 90);修改成camera.setDisplayOrientation(90);

3、改变完方向你会发现方向改变了可是分辨率会变得很低，接下来就是优化了

（1）首先在类CameraManager.java中

把 rect.left = rect.left \* cameraResolution.x / screenResolution.x;

rect.right = rect.right \* cameraResolution.x / screenResolution.x;

rect.top = rect.top \* cameraResolution.y / screenResolution.y;

rect.bottom = rect.bottom \* cameraResolution.y / screenResolution.y;

替换成

rect.left = rect.left \* cameraResolution.y / screenResolution.x;

rect.right = rect.right \* cameraResolution.y / screenResolution.x;

rect.top = rect.top \* cameraResolution.x / screenResolution.y;

rect.bottom = rect.bottom \* cameraResolution.x / screenResolution.y;

（2）然后是在DecodeHandler类中的方法private void decode(byte[] data,int width,int height){｝

中添加byte[] rotatedData = new byte[data.length];

for (int y = 0; y < height; y++) {

for (int x = 0; x < width; x++)

rotatedData[x \* height + height - y - 1] = data[x + y \* width];

\*具体内容可见附件

（3）再就是CameraConfigurationManager类中的方法void initFromCameraParameters(Camera camera){｝中添加如下代码：

Point screenResolutionForCamera = new Point();

screenResolutionForCamera.x = screenResolution.x;

screenResolutionForCamera.y = screenResolution.y;

// preview size is always something like 480\*320, other 320\*480

if (screenResolution.x < screenResolution.y) {

screenResolutionForCamera.x = screenResolution.y;

screenResolutionForCamera.y = screenResolution.x;

注： 到这就完成了。如果还有问题，只需下载附件直接替换相应文件即可； (直接替换很可能不行)

还有一个最后存在的问题，在C:\Users\Administrator\AppData\Roaming\Tencent\Users\1601681656\QQ\WinTemp\RichOle\}ZA28@N$]%W{7~@IA3I1YRP.jpg下编译扫描窗口正常（竖屏），但打包后在DELPHI XE7下，扫描时提示是英文（标题是中文），不知为何？

以上修改之处，是本人环境下完成。

感谢[台北]Q麵（2609815930）、[龟山]阿卍(1467948783) 、[北京]老猫(1765535979)，还有许多不知名的作者。