

Windows Open JDK 替代方案研究

前陣子分享過 [Oracle 追討 Java 授權費議題的研究心得](#)，原以為誤用同綁安裝的進階商業功能才是主要陷阱，但依最近蒐集到的情資，發現對企業用戶而言，「General Purpose Computing」才是天大的坑。依據 Binary Code License (BCL)，Java SE 8 只可免費用於「General Purpose Computing」，用於 Embedded Device 或 Other Computing Environment 就得付費。([參考](#) The current version of Java - Java SE 8 - is free and available for redistribution for **general purpose computing**. Java SE continues to be available under the Oracle Binary Code License (BCL) free of charge. JRE use for **embedded devices** and **other computing environments** may require a license fee from Oracle.) 至於在企業內部的個人電腦安裝及執行 Java SE 是否屬於 General Purpose Computing 用途？BCL 主張「The use of the software in systems and solutions provides dedicated functionality」就不能視為 General Purpose Computing，除非鬧上法院，是不是 General Purpose Computing 誰說了算？你我的膝蓋都知道答案。

對企業而言，遠離付費爭議的最好方法是離 Oracle Java SE 愈遠愈好，網路上很多人推薦改用 [OpenJDK](#)，但 OpenJDK 官網只針對主流 Linux 提供安裝套件，建議 Solaris、Mac OS X、Windows 使用者去找 Oracle。(啊啊啊啊 ~)

爬文找到一些 Windows 使用 Open JDK 的解決方案：

1. RedHat 版

RedHat 從 2016 年 6 月開始提供 [Windows 版 Open JDK](#)，使用者可[下載](#) MSI 在 Windows 安裝，其初衷是方便開發者在 Windows 開發與測試 JBoss Middleware 相關程式，下載前需先填寫資料註冊開發者會員，程序繁瑣了點。

2. 開源社群編譯版本 ojdkbuild

由開源社群熱心維護：<https://github.com/ojdkbuild/ojdkbuild> 可直接下載安裝，且持續定期更新。

3. JDK8 Reference Implementation

官方提供的 JDK8 實作範例：<https://jdk8.java.net/java-se-8-ri/>，Linux、Mac OS X、Solaris 各平台都有，分 GPL 與 BCL 兩種授權。

4. Azul System Zulu

Azul System 這家公司推出基於 Open JDK 開發的開源 [Zulu](#)，標榜通過官方 TCK 認證，與 Java SE 完全相容，開放免費下載及使用，Azul 公司則靠技術支援服務營利。

經評估與測試，RedHat 版需註冊才能下載有點麻煩；Reference Implementation 只能算是 PoC 不會持續發展；開源社群版安裝檔下載與安裝都很方便，唯一的小缺點是 MSI 欠缺數位簽章可能被資安單位挑剔；Zulu 通過 TCK 認證且有廠商背書，感覺更可靠，但商業產品多少會隱含授權政策改變的風險。

【2017-03-11 更新】感謝蘇國鈞老師補充：「Reference Implementation」在 Java 這邊習慣上翻作「參考實作」，簡稱 RI。因為 Java 定義的規格 (JSR) 大家都可以實作，但是收費與否、License 限制各有不同，所以 RI 基本上是一個免費、提供大家學習與實作的一個參考版本。而且很多 RI 實作的嚴謹程度不只是 Proof of Concept，其實是做到 Production-Ready，並且是持續改進的。很多商業/非商業版本的 Application Server，因為必須實作出某些規格才能夠取得 Java EE 認證，或者是不想要/不需要重複發明輪子，常常也都會引用 RI，再加以擴充，來當作自己的實作版本。

最後選擇 ojdkbuild 版，下載 MSI 安裝完畢，小試了 HelloWorld，編譯與執行都沒問題：

```
E:\Java>copy con HelloWorld.java
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World");
    }
}
^Z
複製了      1 個檔案。

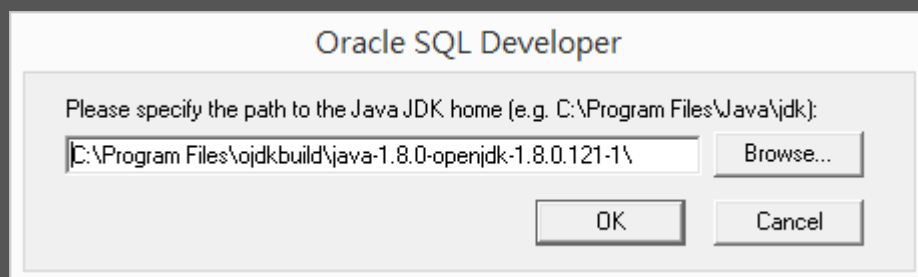
E:\Java>javac HelloWorld.java

E:\Java>java HelloWorld
Hello, World

E:\Java>java -version
openjdk version "1.8.0_121-b13-openjdkbuild"
OpenJDK Runtime Environment (build 1.8.0_121-b13-openjdkbuild-b13)
OpenJDK 64-Bit Server VM (build 25.121-b13, mixed mode)

E:\Java>
```

接著來測試複雜一點的，我刻意挑戰了 Oracle 自己的 Oracle SQL Developer，選擇下載不含 JDK8 的安裝檔。在沒有安裝 Oracle Java SE JDK 的狀況下，程式啟動時會詢問 JDK 位置，將其指向 ojdk-build 的安裝路徑（留意路徑不需包含 bin）：



薑薑薑薑 ~ Oracel SQL Developer 可以使用 ojdkbuild 版 JDK 執行沒問題。（補充：有些程式需依賴 [JAVA_HOME 環境變數](#)，ojdkbuild 不會主動設定，必要時需手動補上）



連 Oracle SQL Developer 用 ojdkbuild 版 JDK 都能過關，使用 Open JDK 執行 Java 應該不用太擔心。比較麻煩的是網頁內嵌 Java Applet 的應用情境，Oracle Java SE 提供與 IE、Firefox、Safari 等瀏覽器整合的 Plugin，如要改用 Open JDK 得自己想辦法。[IcedTea-Web](#) 計劃以自由軟體形式推出允許瀏覽器執行 Java 的套件，但很遺憾沒有現成的 Windows 版實作（主要是 Firefox in Linux），相關研究極少，在 Github 找到一個未成形的 [IcedTea-Web for Windows 專案](#)，幫助不大，Zulu 也坦承沒有加入支援的計劃，加上 Java Applet 技術目前已是風中蠟燭殘燭來日無多（Chrome 很阿莎力地從 42 版就 [不再支援](#)），不指望有人再投注心力，這塊看來是無解了。要在 Windows 續用 Java Applet，只能回歸 Oracle Java SE，或許也可當成一項促使 Java Applet 開發商更快轉向 HTML5 的理由吧！

一開始順順利利解掉大半問題，最後卻在 Java Applet 挨了一記回馬槍，好有吃鍋貼的感覺～（補聲暗）

崇尚自由開放的 Java 被 Oracle 吹皺一池春水，而 C#/.NET 則逐步走向自由開放，還跨了平台，世界真奇妙。

歡迎推文分享：

推到噗浪

Published 10 March 2017 10:18 PM 由 [Jeffrey](#)

Filed under: [Java](#)

Views: 8,326

意見

唯 said on 10 March, 2017 06:57 PM

隔壁 android studio 也直接改用 Open JDK 了所以我想在相容性的部分應該不太需要擔心