

ダブルクリックで起動する JavaFXアプリケーションJAR

第8回 JavaFX勉強会 2012-11-30

高橋 徹 (@boochnich & Java読書会)

発表者について

@boochnich

<http://d.hatena.ne.jp/torutk/>

- Java GUI歴
 - jdk1.0時代はAWT
 - jdk1.1時代はAWT、Netscape IFC、Swing
 - jdk1.2以降はSwing、Java2D、(Java3D)
- Java読書会BOF代表
 - 1998年～毎月1回開催、通算164回、24冊目

<http://www.javareading.com/bof/>

はじめに

- JavaFXアプリケーションの起動方法

- スタンドアロン・プログラム

- Webスタート

- Webページに埋め込み

- セルフ・コンテナメント・アプリケーション

javafxpackagerでスタンドアロン・プログラム
(実行可能JAR)を作成する話です。

アジェンダ

- JDK 7u9の中にJavaFX 2.2.3
- NetBeansでJavaFXアプリケーションを作成
- コマンドでダブルクリック起動JARを作成
- (おまけ) OS固有のインストーラを作成

JDK 7u9の中のJavaFX 2.2.3

C:¥Program Files¥Java¥jdk1.7.0

```
+-- bin
|   +-- javafxpackager.exe
+-- jre
|   +-- lib
|       +-- javafx.properties
|       +-- jfxrt.jar
+-- lib
    +-- ant-javafx.jar
    +-- javafx-doclet.jar
```

(Windows OSの例)

JDKのディレクトリツリー中にJavaFX関連ファイルが含まれています。

(抜粋につき、すべてのJavaFX関連ファイルを記載してはおりません)

NetBeansでJavaFXアプリケーション(1/5)

- NetBeansで新規プロジェクト[JavaFXアプリケーション]を選択し、雛形を作成
- プレゼンなので文字を大きく
 - CSSでチャレンジしてみます
- NetBeans上でビルドして実行



NetBeansでJavaFXアプリケーション(2/5)

- ファイルエクスプローラでJARをダブルクリック



- コマンドプロンプトからjava -jarコマンド実行



NetBeansでJavaFXアプリケーション(3/5)

- ソースを自分でコンパイル、JAR化してみる

```
>javac -cp "C:\Program Files\java\jdk1.7.0\jre\lib\jfxrt.jar"  
-d build\classes -encoding UTF-8 src\hellojfxug\HelloJfxug.java
```

```
>jar cvfm Hello.jar hello.mf -C build\classes .
```

マニフェストが追加されました

hellojfxug/を追加中です(入=0)(出=0)(0%格納されました)

hellojfxug/HelloJfxug\$1.classを追加中です(入=944)(出=539)(42%収縮されました)

hellojfxug/HelloJfxug.classを追加中です(入=1413)(出=864)(38%収縮されました)

hellojfxug/HelloJfxug.cssを追加中です(入=302)(出=214)(29%収縮されました)

```
>java -jar Hello.jar
```

```
Exception in thread "main" java.lang.NoClassDefFoundError: javafx/app  
lication/Application
```

```
    at java.lang.ClassLoader.defineClass1(Native Method)
```

```
    at java.lang.ClassLoader.defineClass(ClassLoader.java:791)
```



残念、動きませんでした。クラスパス指定が不足しています。

NetBeansでJavaFXアプリケーション(4/5)

- 違いはなにか？

- NetBeansの生成したJARと比較してみる



```
com/javafx/main/Main$1.class
com/javafx/main/Main$2.class
com/javafx/main/Main.class
com/javafx/main/NoJavaFXFallback.class
```

NetBeansの生成したJAR
にだけ含まれるクラスファ
イル群

```
Manifest-Version: 1.0
JavaFX-Version: 2.2
implementation-vendor: torutk
implementation-title: HelloJjfxug
implementation-version: 1.0
JavaFX-Application-Class: hellojjfxug.HelloJjfxug
JavaFX-Class-Path: HelloJjfxug.jar
Created-By: JavaFX Packager
Main-Class: com/javafx/main/Main
```

NetBeansの生成したJAR
のMANIFEST.MF内容

NetBeansでJavaFXアプリケーション(5/5)

- `-Djavafx.verbose=true` を指定しJARを起動する

```
>java -Djavafx.verbose=true -jar dist\HelloJjfxug.jar
```

```
: (中略)
```

```
1) Try existing classpath...
```

```
==== URL list
```

```
file:/C:/work/HelloJjfxug/dist/HelloJjfxug.jar
```

```
====
```

```
2) Try javafx.runtime.path property...
```

```
3) Look for cobundled JavaFX ... [java.home=C:\Program Files\Java\jdk1.7.0\jre
```

```
==== URL list
```

```
file:/C:/work/HelloJjfxug/dist/HelloJjfxug.jar
```

```
file:/C:/Program%20Files/Java/jdk1.7.0/jre/lib/jfxrt.jar
```

```
file:/C:/Program%20Files/Java/jdk1.7.0/jre/lib/deploy.jar
```

```
file:/C:/Program%20Files/Java/jdk1.7.0/jre/lib/plugin.jar
```

```
file:/C:/Program%20Files/Java/jdk1.7.0/jre/lib/javaws.jar
```

```
====
```

```
Try calling Class.forName(hellojjfxug.HelloJjfxug) using classLoader = java.net.
```

```
URLClassLoader@06f404c51
```

```
found class: class hellojjfxug.HelloJjfxug
```

```
: (後略)
```



ダブルクリックで起動するJARの作り方

- javafxpackagerを使うと、実行可能JARを作成できます。



```
>javafxpackager -createjar -nocss2bin  
-appclass hellojfxug.HelloJfxug  
-srcdir build\classes  
-outdir .  
-outfile HelloJfxug-0.1
```

Applicationクラスを継承したmainメソッドを持つクラスを指定

-srcdirですが、JARに含めるクラスファイルのあるディレクトリを指定

```
> java -jar HelloJfxug-0.1.jar
```

JARファイル名(拡張子不要)を指定

コマンドリファレンス

<http://docs.oracle.com/javafx/2/deployment/javafxpackager001.htm>

情報源

- 「Oracle Blogs 日本語のまとめ」より
[JavaFX] Native packaging for JavaFX
 - <http://orablogs-jp.blogspot.jp/2012/06/native-packaging-for-javafx.html>
- Oracle JavaFXドキュメント
”Deploying JavaFX Applications”より
 - <http://docs.oracle.com/javafx/2/deployment/jfxp-ub-deployment.htm>
5.5 Create the Main Application JAR File など

まとめ

- JavaFXには、アプリケーションのインストール・実行・更新の仕組みがあります。
- NetBeans 7.3ではプロジェクトの設定のみで簡単に利用できます。
- `javafxpackager`コマンド(またはAntタスク)が提供されています。
- eclipse派な人には`e(fx)cclipse`プラグインがあります。

おまけ MSI形式の作成

- javapackagerでWindows MSI形式のインストーラを作成可能です。
 - 要WiX(Windows Installer XML)・・・無償
- NetBeans 7.3 beta2でも可能です。
- 詳しくはブログで
<http://d.hatena.ne.jp/torutk/20121119>

おまけ2 RPM形式の作成

- javapackagerでLinux RPM形式のインストーラを作成可能です。
 - rpmbuildコマンドが実行できること
- NetBeans 7.3 beta2でも可能です。

サンプルプログラム注釈

CSSファイルの読み込み(ソースファイルと同じ場所に置く場合)

```
Scene scene = new Scene(root, 300, 250);
scene.getStylesheets().add(
    this.getClass().getResource("HelloJjfxug.css").toExternalForm());
```

CSSファイルの記述(Buttonクラスのフォント設定)

```
.button {
    -fx-font-size: 24;
    -fx-font-family: Meiryo sans-serif;
}
```

NetBeansでJavaFXアプリケーション(補1)

- `com.javafx.main.Main`クラスを調べる
 - システムプロパティ`java.home` 
 - `URLClassLoader`でApplication継承クラスをロードし`main`メソッドを実行している。

```
ClassLoader urlClassLoader = new URLClassLoader(urls, null);
:
Thread.currentThread().setContextClassLoader(urlClassLoader);
:
Class appClass = Class.forName(appName, false,
    Thread.currentThread().getContextClassLoader());
:
Method mainMethod = appClass.getMethod("main", new Class[] { new
String[0].getClass() });
:
mainMethod.invoke(null, new Object[] { args });
```

ねこび～ん

- ねこび～んのアイコン

<http://ja.netbeans.org/nekobean/>

- 泣き声

<http://koukaongen.com/>