



MOVING JAVA FORWARD

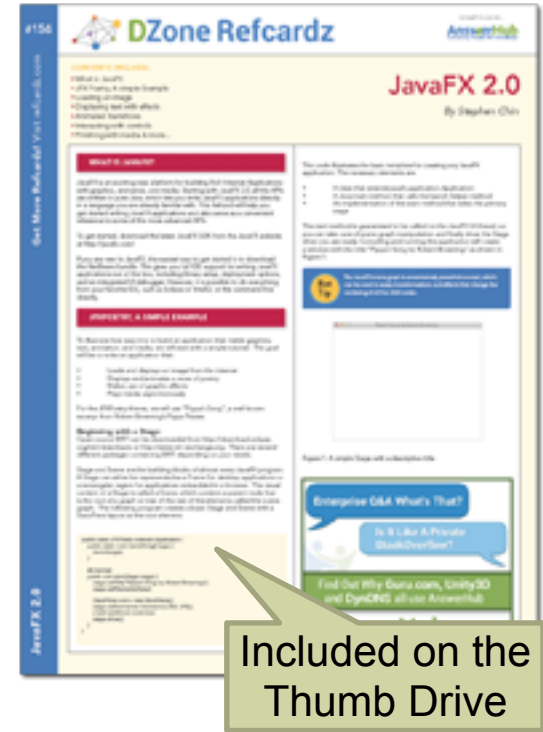
ORACLE®

JavaFX 2 – A Java Developer's Guide

Presented by Stephen Chin <[@steveonjava](#)>

About the Presenter: Stephen Chin

- Oracle Java/JavaFX Evangelist
- Authored several books, most recently Pro JavaFX 2
- Wrote the JavaFX 1 & 2 Refcard
- Blog: steveonjava.com
- Twitter: [@steveonjava](https://twitter.com/steveonjava)
- E-mail: stephen.chin@oracle.com



Safe Harbor Statement

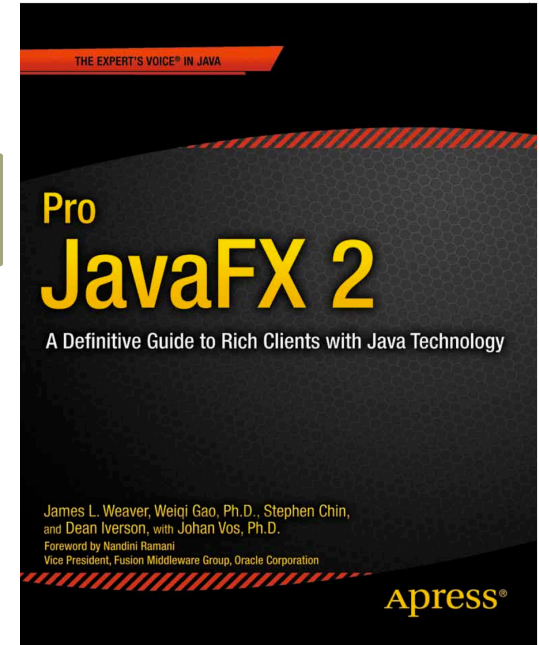
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.



Lab Resources

- Visit ProJavaFX2.com to download Chapter 1 and all code examples
- JavaFX site: oracle.com/javafx
- NetBeans site: netbeans.org
- fxexperience blog: fxexperience.com
- Jim Weaver's JavaFX blog: JavaFXpert.com
- My JavaFX blog: steveonjava.com

This is on the Drive Too!



What I want you to take away

1. The JRE with JavaFX enables graphically rich, fast performing applications
2. Developing JavaFX apps is natural and fun
3. Lots of great resources exist to help you get started with JavaFX
4. You should begin playing with JavaFX now!



1. Getting Started with JavaFX

- Bringing back rich-client Java
- Learning the history of JavaFX
- Obtaining JavaFX tools and resources
- Building and running Hello EarthRise
- Reviewing the structure of Hello EarthRise



Bringing back rich-client Java

- The web, originally intended for sharing hypertext-linked resources, has been force-fit into being a rich-client platform
- The JRE with JavaFX is a much better client platform, enabling graphically rich, fast performing applications
- See Chapter 1 *JavaFX Can't Bring Rich-Client Java Back by Itself* section



Learning the history of JavaFX

- The brainchild of Chris Oliver while at SeeBeyond, which was acquired by Sun
- JavaFX Script language and platform announced at JavaOne 2007
- Oracle implemented JavaFX 2 as an API for Java, announced at JavaOne 2010
- JavaFX 2.1 GA released for JavaOne India 2012
 - Windows and OS/X now GA, Linux GA soon
- See Chapter 1 *A Brief History of JavaFX* section



Obtaining JavaFX tools and resources

- Visit oracle.com/javafx and click **Download**
- If you are on:
 - Windows/Mac – Grab the co-bundled Java SE 7u5, JavaFX 2.1.1 and NetBeans 7.1.2
 - Linux – Grab the JavaFX 2.2 Developer Preview
- Note: Java SE 7u4+ bundles JavaFX 2.1

And of course I put these on...

Building and running Hello EarthRise

- Follow the steps in Chapter 1 *Building and Running the Program with NetBeans* section



Demo



Review the structure of Hello Earthrise

- Application class and overridden `start ()`
- Declarative code that defines the UI
- Using the `Stage` class
- Using the `Scene` class
- Displaying images
- See Chapter 1 *Understanding the Hello Earthrise Program* section

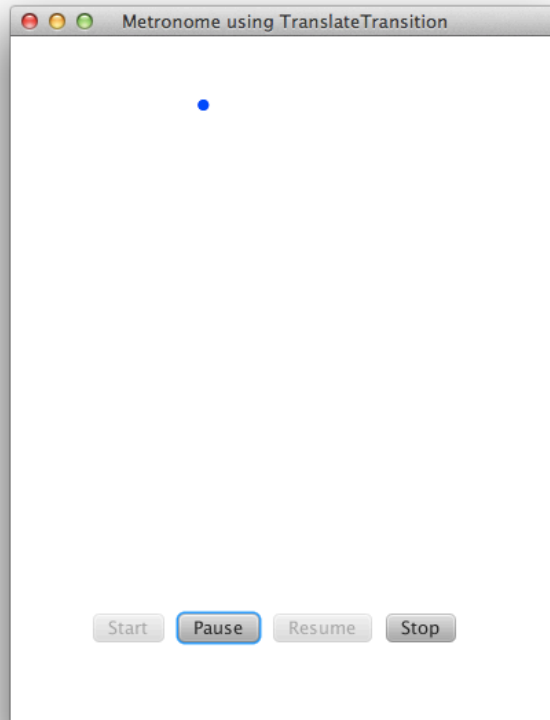
Code Walkthru



2. Creating a User Interface in JavaFX

- Demo the MetronomeTransition app
- Examine the MetronomeTransition code
- Expressing the UI in a declarative manner
- Binding the UI to properties in the model
- Animating nodes

Demo the MetronomeTransition app



- MetronomeTransition code is in the 8727_ch02code folder of the Pro JavaFX 2 code download bundle

Demo



Examine MetronomeTransition code

- Expressing the UI in a declarative manner
- Using layout classes
- Binding the UI to properties in the model
- Animating nodes



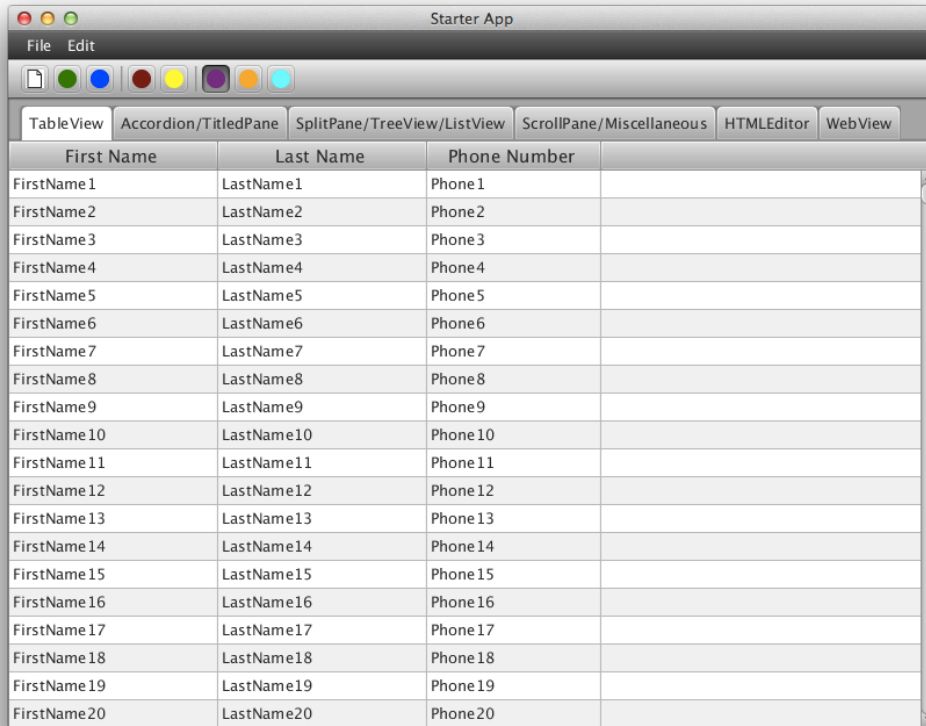
Code Walkthru



3. Using the JavaFX UI Controls

- Demo the StarterApp example
- Examine the StarterApp code
- Survey UI controls in StarterApp
- Leveraging the UI controls in code
- Associating a JavaFX CSS style sheet
- Defining model classes

Demo the StarterApp example



The screenshot shows a JavaFX application window titled "Starter App". It has a menu bar with "File" and "Edit". Below the menu bar is a toolbar with icons for file operations. The main content area displays a table with the following data:

First Name	Last Name	Phone Number	
FirstName1	LastName1	Phone 1	
FirstName2	LastName2	Phone 2	
FirstName3	LastName3	Phone 3	
FirstName4	LastName4	Phone 4	
FirstName5	LastName5	Phone 5	
FirstName6	LastName6	Phone 6	
FirstName7	LastName7	Phone 7	
FirstName8	LastName8	Phone 8	
FirstName9	LastName9	Phone 9	
FirstName10	LastName10	Phone 10	
FirstName11	LastName11	Phone 11	
FirstName12	LastName12	Phone 12	
FirstName13	LastName13	Phone 13	
FirstName14	LastName14	Phone 14	
FirstName15	LastName15	Phone 15	
FirstName16	LastName16	Phone 16	
FirstName17	LastName17	Phone 17	
FirstName18	LastName18	Phone 18	
FirstName19	LastName19	Phone 19	
FirstName20	LastName20	Phone 20	

- StarterApp code is in the 8727_ch05code folder of the Pro JavaFX 2 code download bundle

Demo



Examine StarterApp code

- Survey UI controls in StarterApp
- Leveraging the UI controls in code
- Associating a JavaFX CSS style sheet
- Defining model classes



Code Walkthru



4. Using the Media Classes

- Demo the VideoPlayer1 example
- Examine the VideoPlayer1 code
- Using the MediaView, MediaPlayer and Media classes
- Using a procedural vs. declarative approach

Demo the VideoPlayer1 example



- StarterApp code is in the 8727_ch08code folder of the Pro JavaFX 2 code download bundle

Demo



Examine VideoPlayer1 code

- Using the MediaView, MediaPlayer and Media classes
- Using a procedural vs. declarative approach

Code Walkthru



5. Deploying JavaFX apps

- Some options are Java WebStart and running as an applet
 - e.g. Icon at well-known web page that invokes TweetBrowser via Java WebStart
- Another good deployment option is creating an installer
 - “there’s an app for that”
 - See related FXexperience.com blog post
 - Packager preview available in JavaFX 2.2 – will be part of the standard JDK



6. Embedding HTML5

- Show the WebView API
- Demo the Java Conference Tour example
- Examine the Java Conference Tour architecture



Displaying HTML in JavaFX

```
public class WebViewTest extends Application {  
    public static void main(String[] args) {  
        launch(WebViewTest.class, args);  
    }  
    @Override public void start(Stage stage) {  
        WebView webView = new WebView();  
        webView.getEngine().load("http://google.com");  
        Scene scene = new Scene(webView);  
        stage.setScene(scene);  
        stage.setTitle("Web Test");  
        stage.show();  
    }  
}
```

Displaying HTML in JavaFX



Calling Javascript from JavaFX

```
String script = "alert('We have got a message, Houston!');";  
eng.executeScript(script);
```

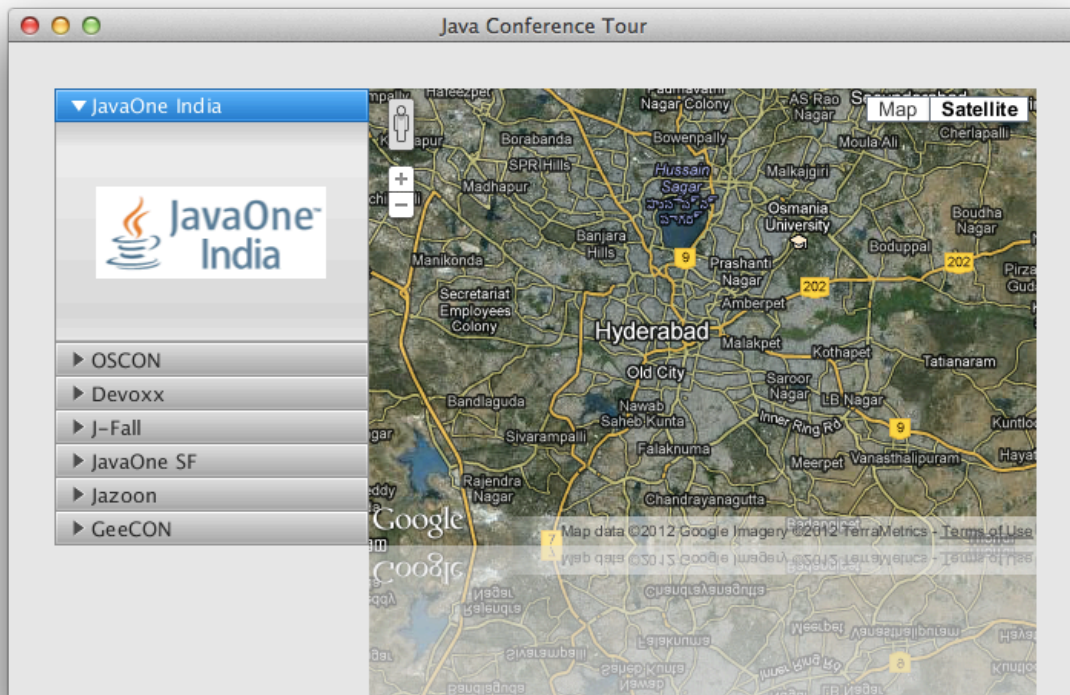
JavaScript to Java Type Conversion

JavaScript	Java
null	null
undefined	“undefined”
number	java.lang.Number (Integer or Double)
string	java.lang.String
boolean	java.lang.Boolean
object	netscape.javascript.JSObject

Responding to Browser Events

- Alert/Confirm/Prompt – Respond to JavaScript user interaction functions
- Resize – Web page moves or resizes the window object
- Status – Web page changes the status text
- Visibility – Hide or show the window object
- Popup – Spawn a second web view/engine

The Java Conference Tour app



Demo



Take-aways from this presentation

1. The JRE with JavaFX enables graphically rich, fast performing applications
2. Developing JavaFX apps is natural and fun
3. You can use the best parts of JavaFX and HTML5 together in the same application
4. Lots of great resources exist to help you get started with JavaFX
5. You should begin playing with JavaFX now!



Lab Time!

<now it's your turn>

