

Intro To JavaFX

Jonathan Giles,
Software Engineer, JavaFX Team
Oracle Corporation

Introduction



MASSEY UNIVERSITY

BE

2003-2006

MSc

2007

SCJP

2009

Introduction



 JFXStudio: sketch, hack, share



Introduction

ORACLE®



Introduction

I live/work/‘play’ in Thames,
New Zealand

But I work for Oracle,
based in California.



Presentation Overview

- Part 1:
 - JavaFX Introduction
 - How to get started developing JavaFX
 - Question time
- Part 2:
 - JavaFX Script Introduction
 - Question time



Note:

This is an overview.
Shout questions at your leisure.
Interrupt me if you need to.

Part One

JavaFX Introduction

What is JavaFX?

- It's a language for creating graphical user interfaces...

What is JavaFX?

- It's for creating graphical user interfaces...

See <http://www.youtube.com/watch?v=hkDDo3yeLnU>

What is JavaFX?

- It's for creating graphical user interfaces...
- Problems:
 - Windows-only
 - Desktop-only
 - Ugly / Not easily styled
- It's freakin' Visual Basic!



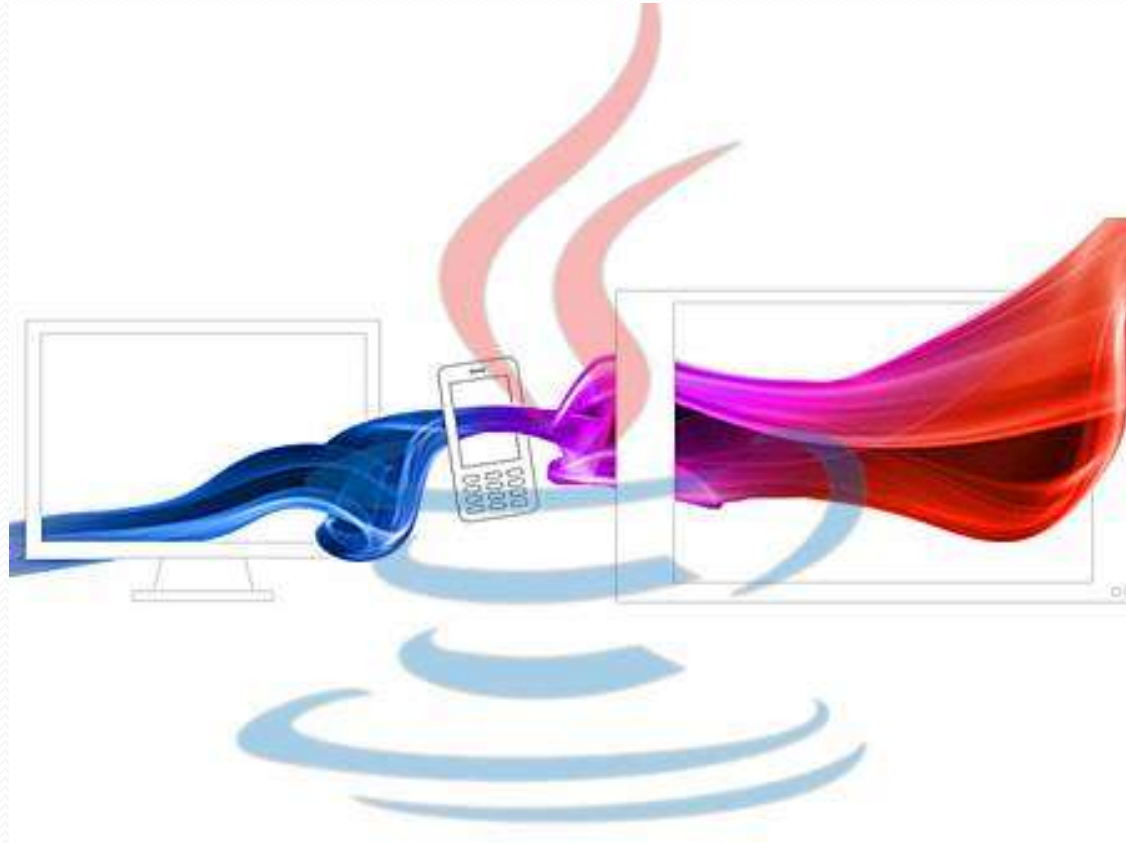
What is JavaFX?

It's three things.



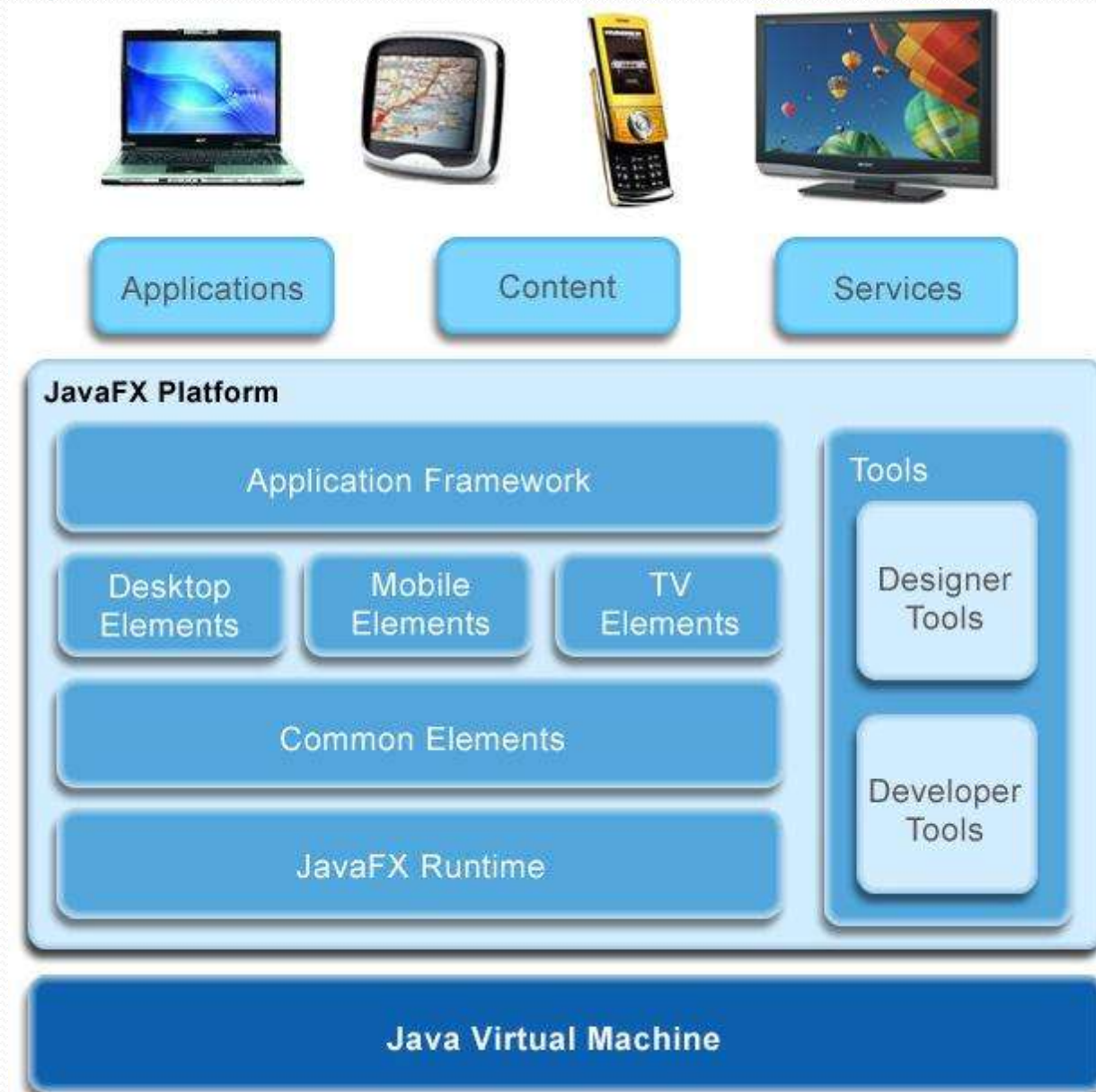
What is JavaFX?

Number one:
Common API
over mobile, TV
and desktop



What is JavaFX?

Number two:
Runtime on top
of the JRE



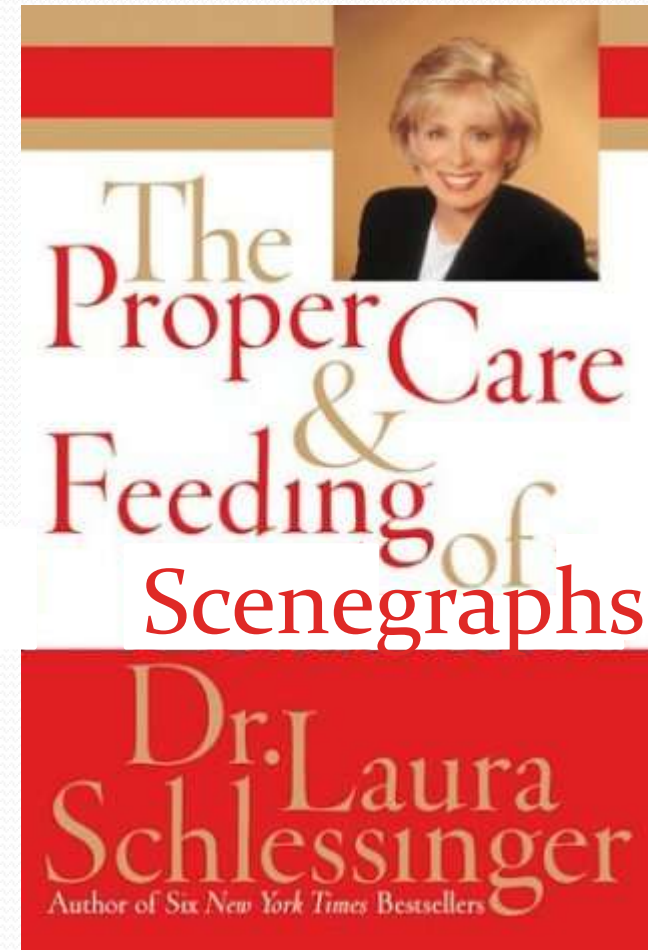
What is JavaFX?

Number three:
JavaFX Script



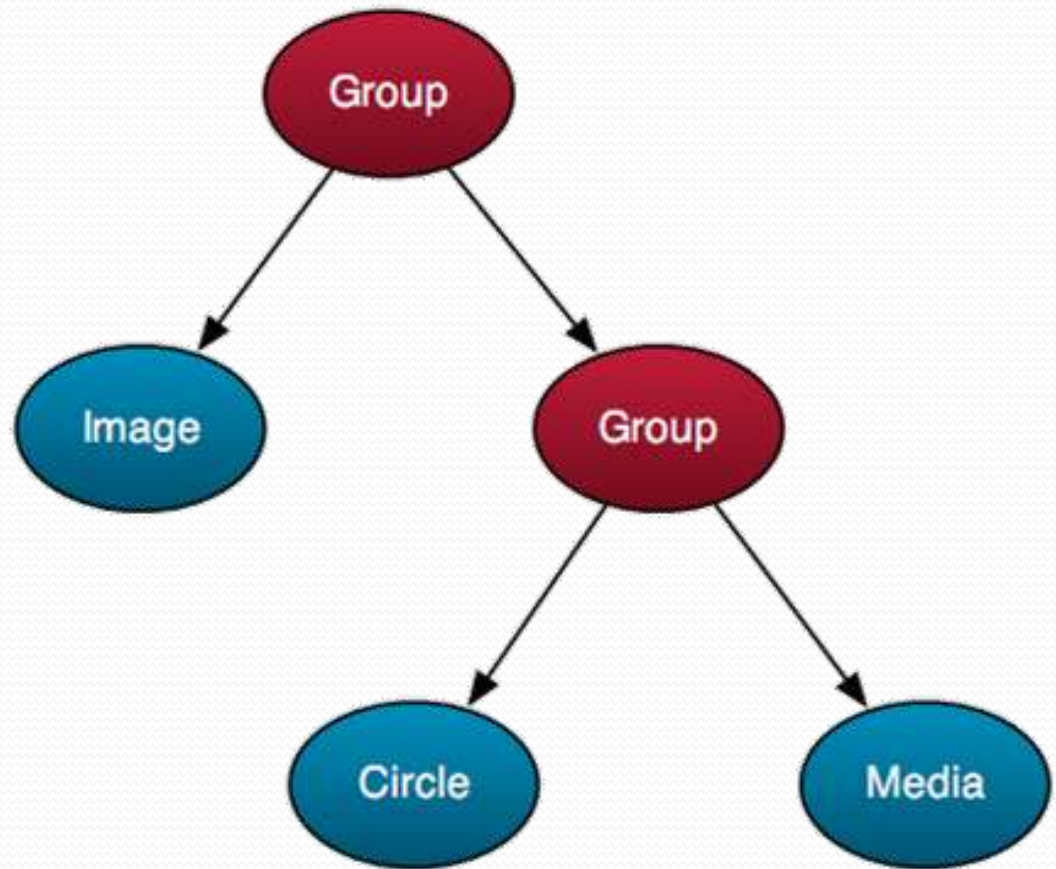
What is JavaFX Script?

“JavaFX Script is a domain specific language for the care and feeding of the scenegraph”



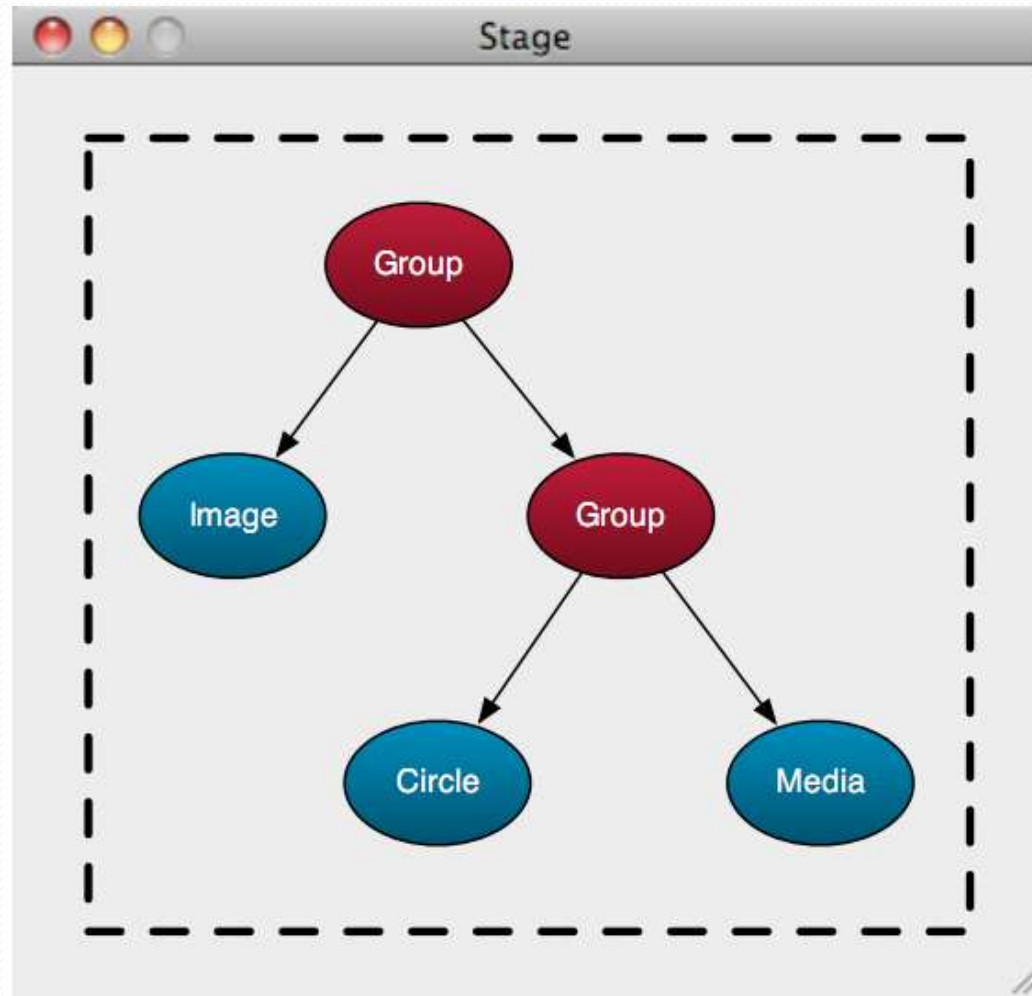
Huh?! Scenegraph

- Data structure representing all visual elements
- Can easily reference any visual element in the scenegraph and manipulate
- Common in games world
- Not as common to desktop developers
 - Used to 'immediate-mode' graphics (e.g. Java2D)



Scenegraph Example

```
Group {  
  content: [  
    ImageView { }  
    Group {  
      content: [  
        Circle { }  
        MediaView { }  
      ]  
    }  
  ]  
}
```



Hello World

```
println("Hello World")
```

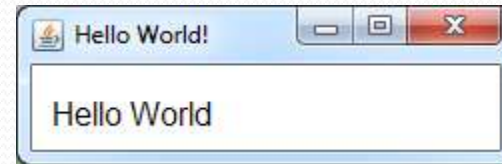


Hello World

```
package helloworld;

import javafx.stage.Stage;
import javafx.scene.Scene;
import javafx.scene.text.Text;
import javafx.scene.text.Font;

Stage {
    title: "Hello World!"
    width: 250 height: 80
    scene: Scene {
        content: [
            Text {
                font : Font { size : 16 }
                x: 10 y: 30
                content: "Hello World"
            }
        ]
    }
}
```



Demo

Scenegraph Manipulation

Why A New Language?

A programming language is not about telling the computer what to do, but instead about expressing the programmer's intent.

Languages differ for this reason.

JavaFX 1.3

- Released last month
- Huge step forward
- Introduced:
 - New controls,
 - Preview of new graphics stack
 - Compiler improvements (memory / CPU)

JavaFX Controls

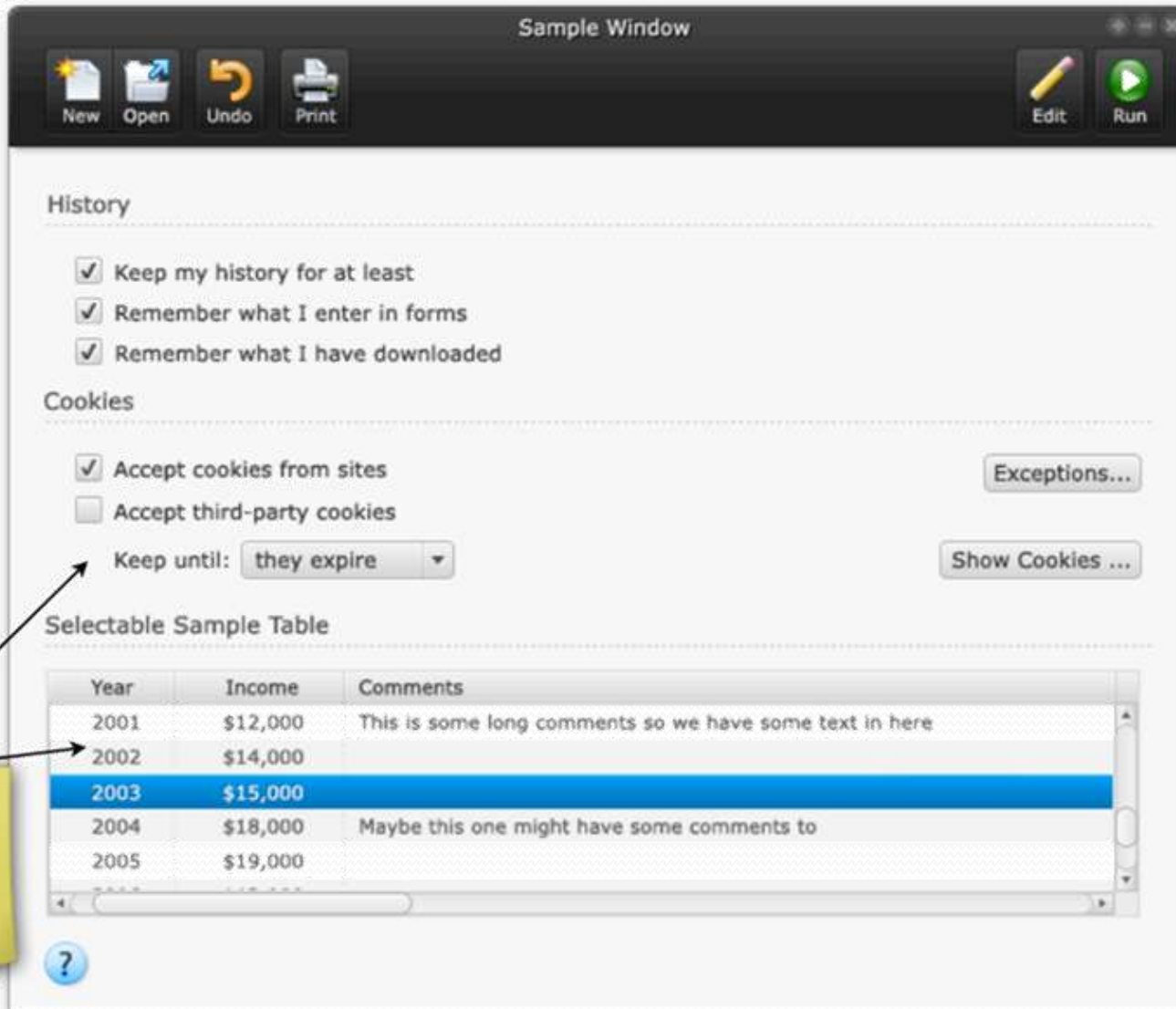
JavaFX 1.2	JavaFX 1.3
Button	Tooltip
ToggleButton	PasswordBox
RadioButton	ChoiceBox
CheckBox	MenuButton
Slider	SplitMenuButton
Label	Horizontal ListView
Hyperlink	ScrollView
ProgressIndicator	Multiline TextBox
ProgressBar	ToolBar (Preview)
TextBox	Menus (Preview)
ListView	Popup (Preview)
ScrollBar	TreeView (Preview)

Demo

JavaFX Controls

CSS Styling

- CSS is our strategy for styling in JavaFX 1.3. If you use our UI controls, you use CSS.
- Caspian is our default stylesheet.
- CSS is fast, and works on mobile, desktop, and tv.
- We stick to the spirit of HTML CSS, but we are not bound by it.

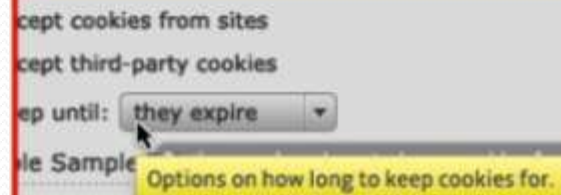


Dark color
buttons on
toolbar

Light color
controls on
almost white
background

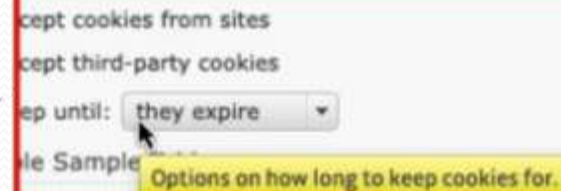
CSS

```
Scene {  
  base-color: #646464  
}  
  
Tooltip {  
  background-color: yellow;  
  cursor: hand  
}
```



cept cookies from sites
cept third-party cookies
ep until: they expire
le Sample
Options on how long to keep cookies for.

```
Scene {  
  base-color: #CBCBCB  
}  
  
Tooltip {  
  background-color: yellow;  
  cursor: hand  
}
```



cept cookies from sites
cept third-party cookies
ep until: they expire
le Sample
Options on how long to keep cookies for.

```
Scene {  
  base-color: #111111  
}  
  
Tooltip {  
  background-color: yellow;  
  cursor: hand  
}
```



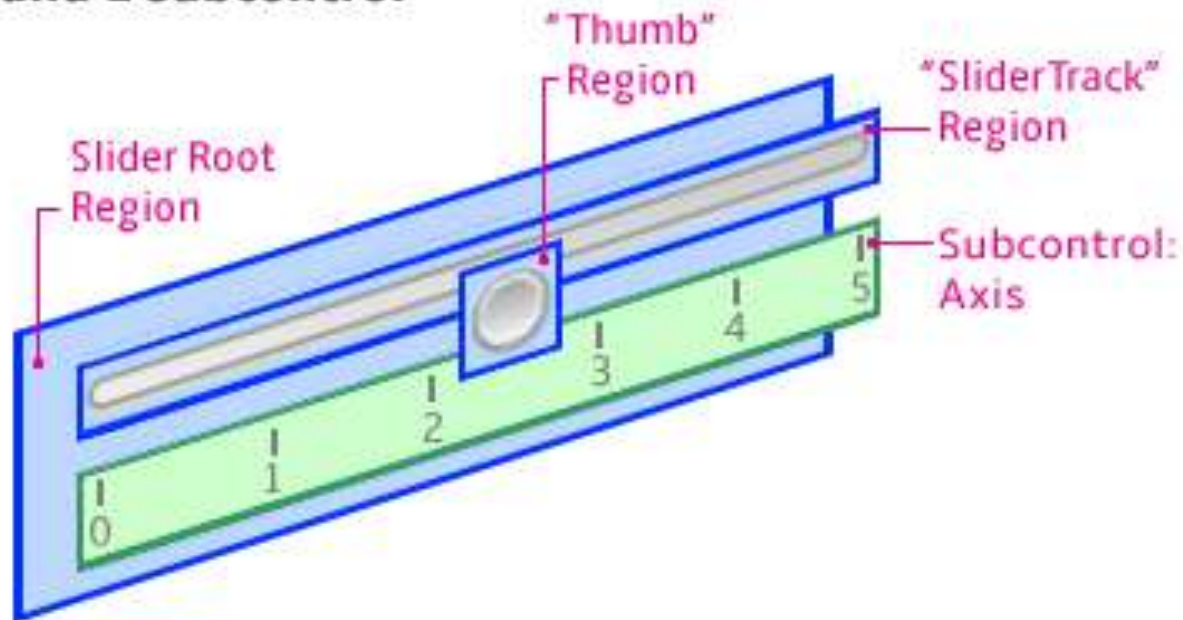
cept cookies from sites
cept third-party cookies
ep until: they expire
le Sample
Options on how long to keep cookies for.

Regions

- In JavaFX 1.3, we rewrote all control skins.
- Broke each skin into stylable parts.
- Region is a Rectangle, with independently rounded corners or any arbitrary path.
- Can have multiple background fills, background images, border strokes and border images.

Regions: Slider

Slider Control Contains 2 Regions and 1 subcontrol



-  Regions
-  Subcontrol: Axis
-  Graphical Content

Getting Started with JavaFX

NetBeans

- Best tool for the job at present
 - However, Eclipse and IntelliJ have (growing) support.
- Download JavaFX plugin from within NetBeans, or download everything you need (including JavaFX SDK), from <http://www.netbeans.org>
- NetBeans Composer plugin
 - RAD for developers

Coming up...

- JavaFX Authoring Tool
 - Primarily for designers
- JavaFX 1.3.x
 - Throughout this year
- JavaFX 1.4
 - Later 2010/2011

End of Part One

Questions?

Part Two

JavaFX Script Deep Dive

What is JavaFX Script?

- Is a compiled language
- Very similar to Java, but somewhat more designer friendly
- Features:
 - Null pointer-less language
 - Expression language
 - Data binding
 - Event triggers
 - Function types
 - Mixin inheritance
 - Type inference
- You can make full use of Java libraries and API.

Type Inference

- Statically typed (not dynamic)
- Every variable has a type, even if type not declared
- Compiler can often figure types out
 - `var x = 5; // Integer`
 - `var cond = true; // Boolean`
 - `function f() { "xyz" } // String`

Data Types

- Primitive types from Java:
 - Boolean, Integer, Long, String, ...
- Number
- **Duration**
- Primitives cannot be null
 - Null string coerced to an empty string
- Void (only as return “type” of a function)

A (Very) Basic Binding Example

```
var x = 0;                // x starts by being set to 0
var y = bind x + 10;      // y starts by being bound to x + 10

x = 1;                    // x changes to 1. y changes to 11
println(y);              // This proves it by printing 11 to the console

x = 42;                   // Again, x is change to 42, and y follows suit
println(y);              // Which is proven when println results in 52 being displayed
```


Sequences

- Similar to Java's arrays, but better.
- Examples:

```
def days = ["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"];
```

```
var nums = [1..100];
```

```
var numsGreaterThanTwo = nums[n | n > 2];
```

```
insert 101 into nums;
```

```
insert 0 before nums[0];
```

```
var squares:Integer[] = for (x in [1..10]) x * x;
```

```
--> [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

Binding

- conditional binding:
 - `def z = bind if (x == y) 1024 else 2048;`
- bind to a sequence:
 - `var squares= bind for (x in [min..max]) x * x;`
- bind to a function:
 - `def value = bind myFunc(x,y)`
- bound functions:
 - `bound function myFunc(x, y) { x * y * z }`

Event Triggers

```
var password = "foo" on replace {  
    println("ALERT! Password has changed!");  
    println("Old Value: {oldValue}");  
    println("New Value: {password}");  
};
```

```
password = "bar";
```

Function Types

```
public var f  
public var f:Integer;
```

```
function sum(x:Integer, y:Integer):Integer { x + y }
```

```
f = sum;
```

```
var result = f(10,10);  
println("result is: {result}");
```

Animation

- Language support for animation / time
 - Duration class
 - Time literals such as 5ms, 5s, 5m, and 5h
 - `var time = 750ms`
- Built in Transitions:
 - Rotate, Fade, Translate, Path, Scale, etc
- Easy to create your own Timelines

Demo:

Animation

Object Oriented

```
class Address {  
    public var street: String;  
    public var city: String;  
    public var country: String;  
}
```

```
Address {  
    street: "1 Main Street"  
    city: "Palmerston North"  
    country: "New Zealand"  
}
```

```
def customer = Customer {  
    firstName: "Jonathan"  
    lastName: "Giles"  
    phone: "0211089038"  
    address:
```

```
}
```

Mixins

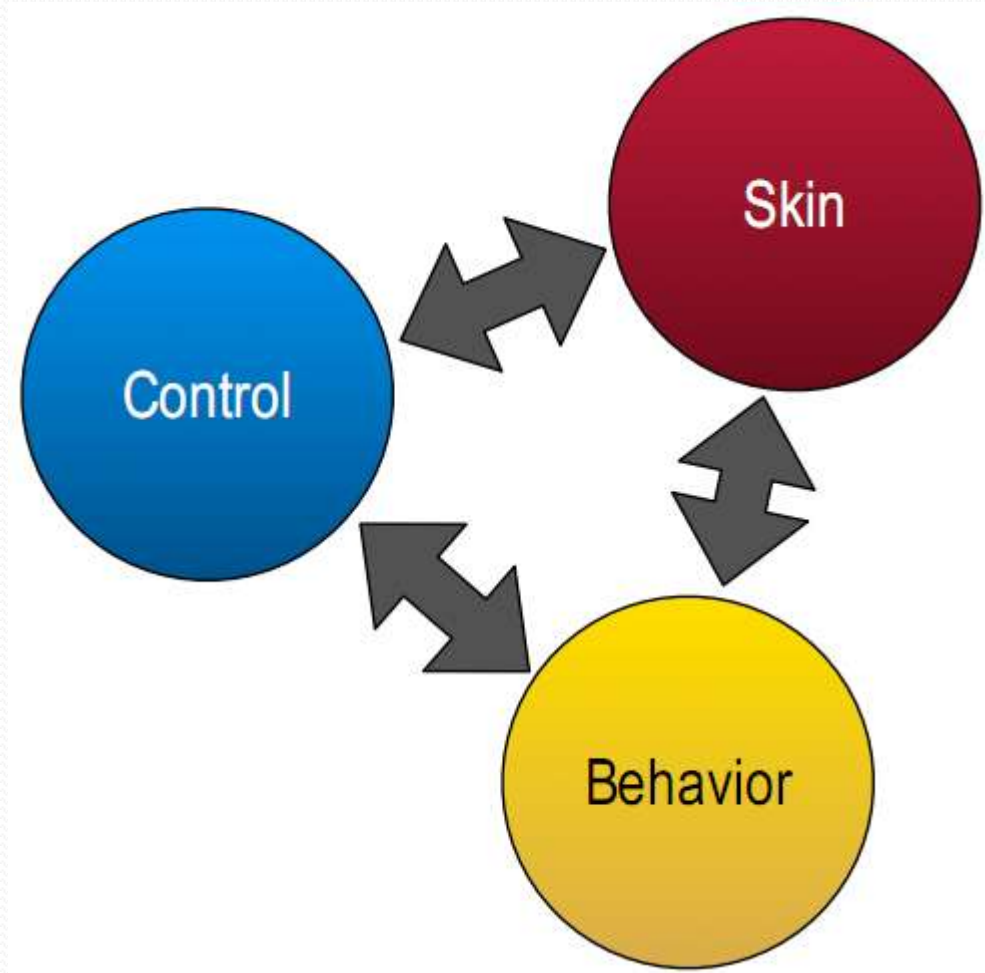
- Mixin classes are like Java interfaces, but...
 - Can include function implementations
 - Can include variable declarations and init values
- A class can extend from any number of mixin classes

```
class A {  
    var x;  
    function f() { ... }  
}
```

```
mixin class M {  
    var y;  
    function g() { ... }  
}
```

```
class B extends A, M {  
    ...  
}
```

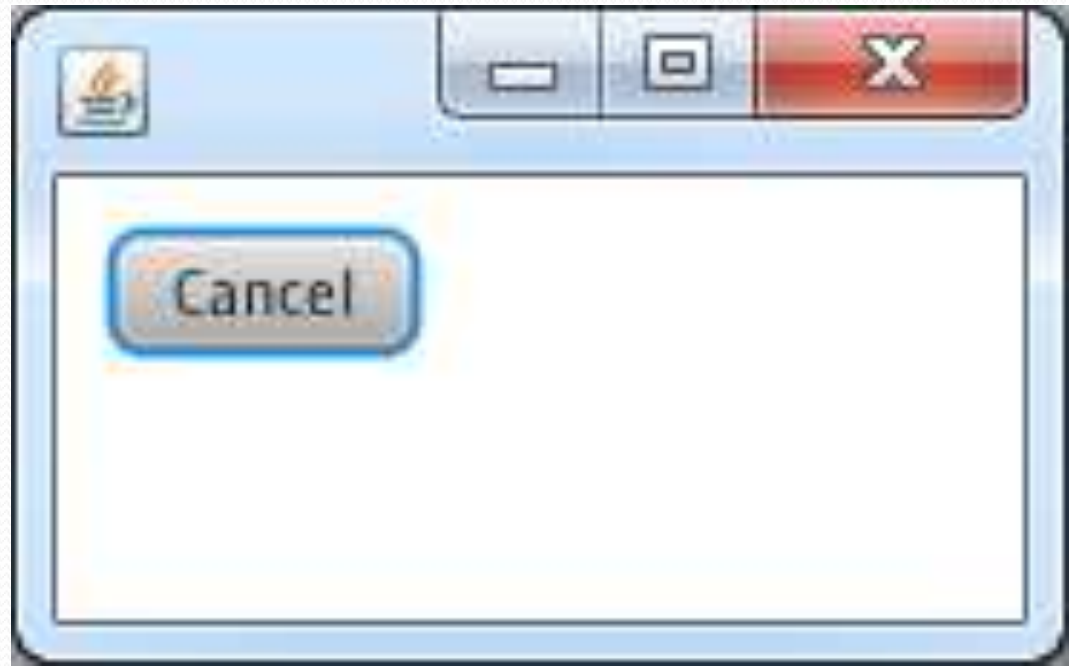

Controls



JavaFX Button

- Example:

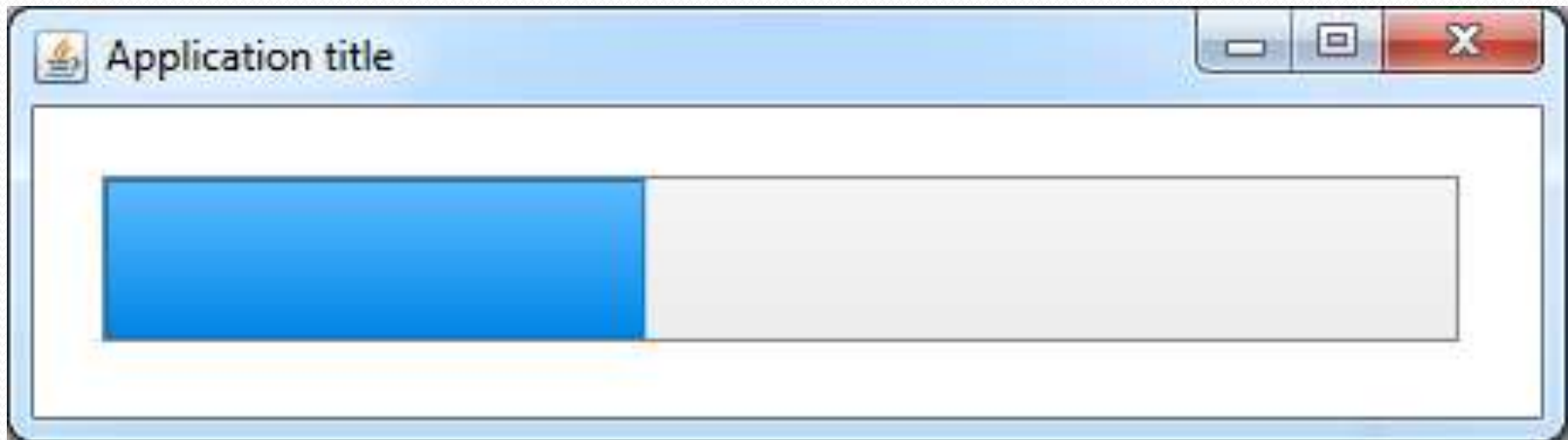
```
Button {  
    text: "Cancel"  
    action: function() {  
        println("I've been clicked!");  
    }  
}
```



JavaFX Progress Bar

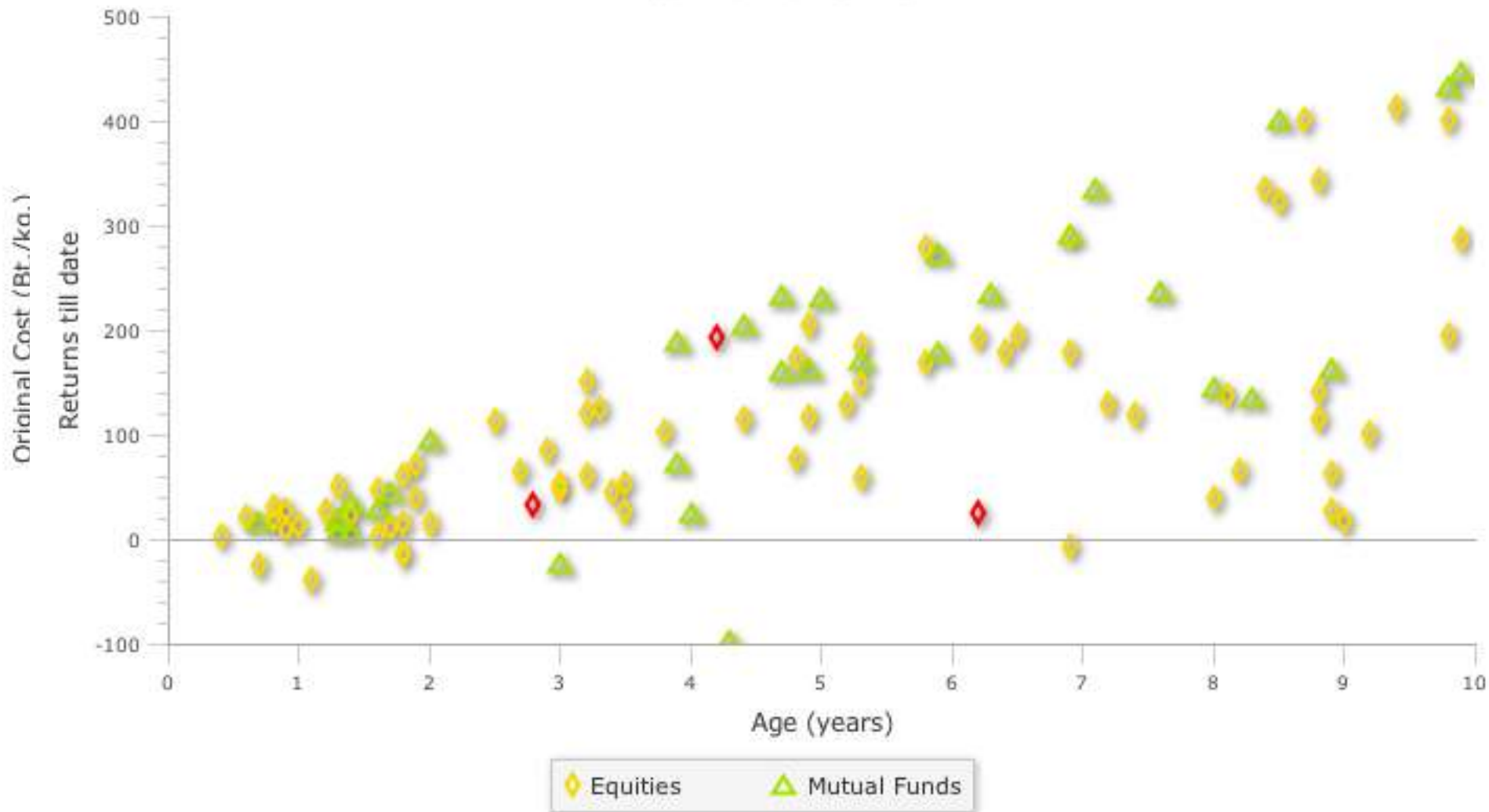
- Example:

```
var task = HttpRequest { .... }  
ProgressBar { progress: bind task.percentDone }
```



Charts

Investment Overview



Demo:

Media

The Future

Prism

What is Prism?

- Presently JavaFX uses Swing/AWT/Java2D.
 - Immediate mode graphics
- Prism is a new scenegraph and graphics stack
 - Retained mode graphics
 - Direct to hardware
 - Just enough for JavaFX needs
- Prism is available in JavaFX 1.3, but not enabled by default.
- Try not to use Swing components in JavaFX for maximum portability.

Demo:

Magic 8 Ball

For more information...

Jonathan.Giles@oracle.com

<http://www.twitter.com/JonathanGiles>

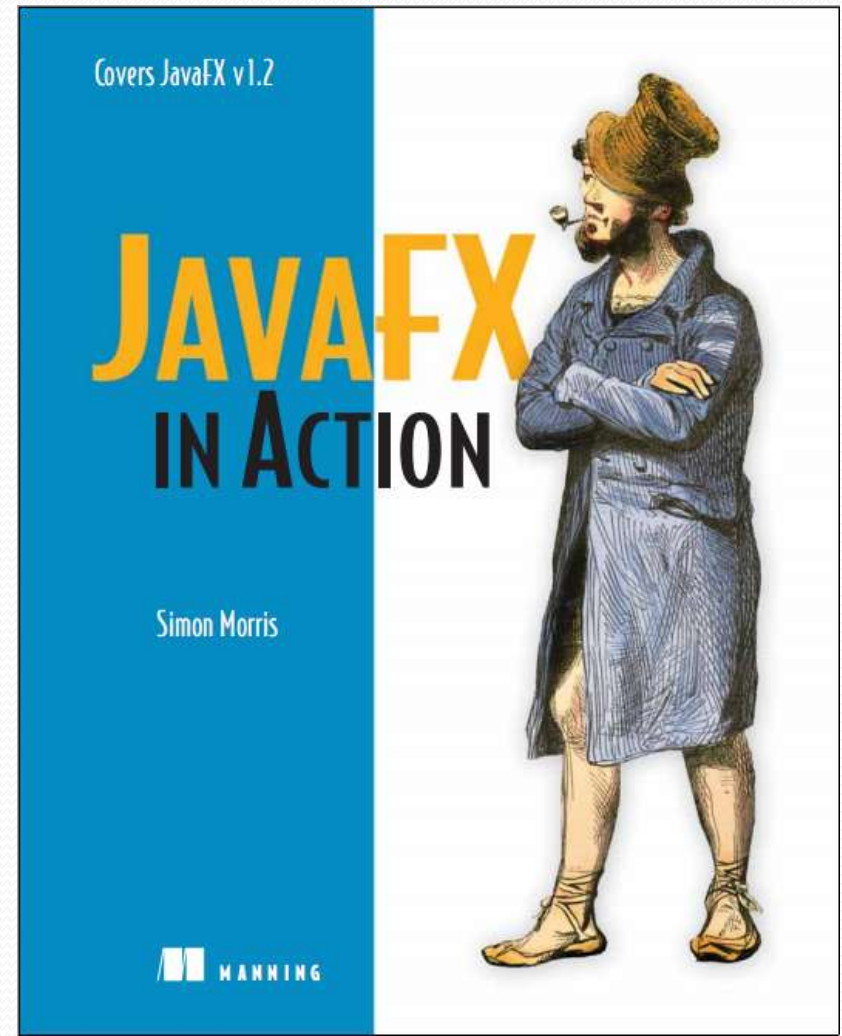
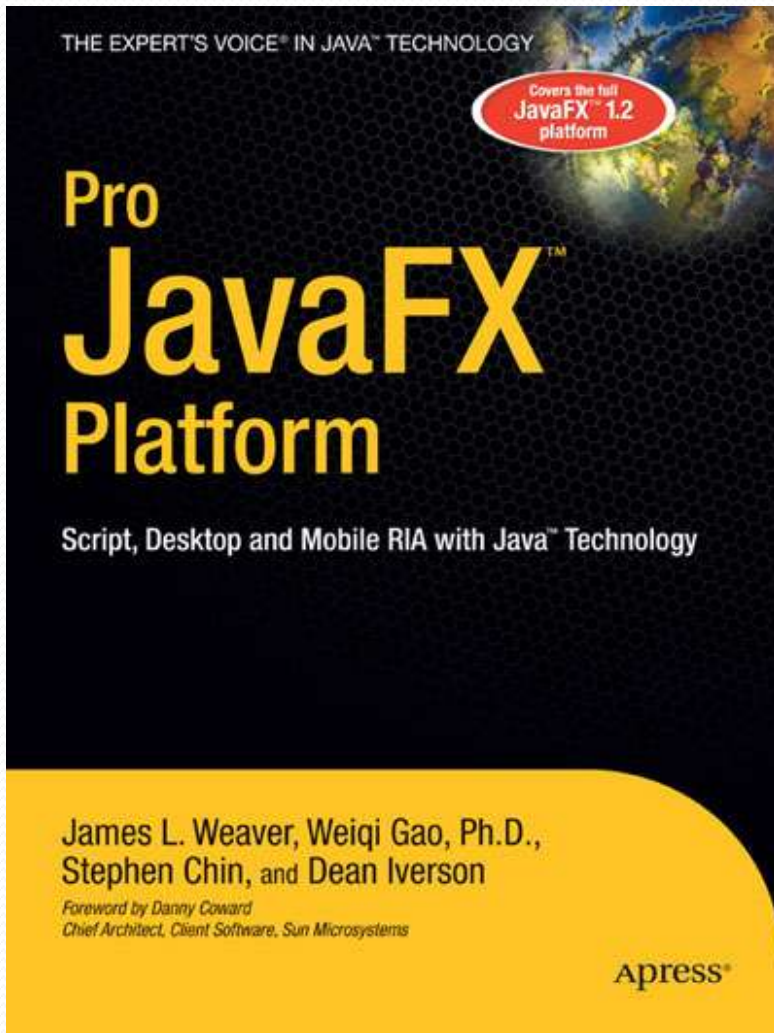
<http://www.JavaFX.com>

<http://www.JonathanGiles.net>

<http://www.fxexperience.com>

<http://www.jfxstudio.org>

So, who wants to learn more?



Questions?

Remember: I'm in the JavaFX controls team.

I don't do marketing-type answers.

But I also can't answer everything,
and I certainly don't know everything.