



Pontifícia Universidade Católica de Minas Gerais
Campus de Poços de Caldas
Departamento de Ciência da Computação - DCC
Curso de Bacharelado em Ciência da Computação

INTRODUÇÃO À PLATAFORMA JAVA FX

Prof. Dr. João Benedito dos Santos Junior

Pontifícia Universidade Católica de Minas Gerais - PUC Minas
Campus de Poços de Caldas
Curso de Bacharelado em Ciência da Computação
Laboratório de Televisão Digital Interativa - TVDILab
<http://tvdilab.inf.pucpcaldas.br>


Outubro de 2014

Orientações para Leitura deste Texto

Este documento foi elaborado com a finalidade de apresentar os conceitos e princípios essenciais do desenvolvimento de aplicações usando a Plataforma JavaFX.


Ao ler o texto, você encontrará algumas marcas  **Leia mais.** . Essas marcas estão colocadas após uma palavra e/ou frase destacada em *itálico*; então, ao encontrar uma marca, esteja atento e procure pesquisar e saber mais sobre a palavra e/ou frase por ela apontada.

1 Introdução à Plataforma JavaFX

JavaFX é uma plataforma, baseada em tecnologia Java, para construção de aplicações ricas , tendo sido projetada para permitir aos desenvolvedores a criação e distribuição de dessas aplicações de forma consistente e transparente em múltiplas plataformas de hardware e software.

A plataforma JavaFX oferece um conjunto de aplicações ricas em aspectos gráficos e manipulação de objetos de mídia, tendo como premissa simplificar o desenvolvimento de aplicações voltadas a ambientes multimídia interativos, tanto aqueles baseados em *desktop* quanto para dispositivos móveis (celulares, *smartphones*, dentre outros) e televisores (Oracle, 2012).

2 Um Breve Histórico sobre JavaFX

Durante a conferência JavaOne'2007, a Sun Microsystems, até então a fabricante das tecnologias Java, apresentou a plataforma JavaFX como novidade aos desenvolvedores de conteúdo e aplicações multimídia ricas em dispositivos móveis, *desktops*, televisores, dentre outros. Inicialmente, consistia da plataforma **JavaFX Mobile** e de uma linguagem de script, denominada JavaFX Script . Após algumas versões, em abril de 2010, foi lançada a plataforma JavaFX 1.3; nesse mesmo ano, foi anunciado que o suporte para a linguagem JavaFX Script seria descontinuado, porém todas suas APIs seriam portadas e liberadas como parte da plataforma JavaFX 2.0, não tendo a necessidade dos desenvolvedores aprenderem uma nova linguagem de *script*, mas sim reutilizar todo o conhecimento adquirido sobre a programação orientada a objetos em Java (Oracle, 2012). Mais recentemente, a plataforma JavaFX foi incorporada à distribuição *standard* da plataforma Java, acompanhando distribuições importantes como é o caso do NetBeans, notadamente a partir da sua versão 8.

Neste contexto, são as principais características da plataforma JavaFX:

- um framework de APIs Java, totalmente orientadas a objeto, para que a plataforma JavaFX permita o desenvolvimento de aplicações pelos desenvolvedores já acostumados com a tecnologia Java, sem necessidade de aprendizado de uma linguagem de *scripts*;
- um novo mecanismo gráfico para lidar com as modernas unidades de processamento gráfico (GPUs), sendo a base deste novo mecanismo o pipeline de aceleração gráfica em hardware, denominado *Prism*;
- um novo mecanismo de mídia (motor multimídia) que suporta reprodução de conteúdo web

multimídia e objetos de mídia contínua em tempo real;

- um componente web que permite a incorporação de conteúdo HTML em uma aplicação JavaFX;
- um plugin que permite a carga (*load*) *applets* JavaFX a partir da estrutura de processamento gráfico do *Prism*;
- uma nova e ampla lista de *widgets* gráficos sofisticados para controle da interface do usuário (UI), que inclui gráficos, tabelas, menus e painéis.

3 Criando uma Aplicação Utilizando JavaFX

As aplicações JavaFX são escritas, essencialmente, em Java, podendo-se utilizar qualquer ambiente integrado de desenvolvimento (IDE), tais como NetBeans, Eclipse, dentre outros.

Para iniciar o desenvolvimento de uma aplicação JavaFX, é necessário instalar o *framework* JavaFX SDK, que acompanha a versão 8 do IDE NetBeans, lançada em 2014; ao instalar essa versão, todo o *framework* JavaFX é instalado conjuntamente, não havendo necessidade de quaisquer *plugins*.

Tomando-se por base um ambiente de desenvolvimento como o NetBeans, para criar um projeto JavaFX, deve-se selecionar o menu *File* e escolher a opção *New Project*. Na categoria *JavaFX*, deve-se escolher a opção *JavaFX Application*, informando o nome que se deseja atribuir ao projeto, finalizando a sua criação. A **figura 1** apresenta um projeto-padrão, denominado *HelloWorld.java*, que pode ser utilizado para compreender a estrutura básica de uma aplicação JavaFX. Segundo Santos Junior (Santos Jr., 2012), toda e qualquer Engine JavaFX começa a executar uma *Application JavaFX* pela instrução **launch**, que significa *execute*. Essa instrução **launch** deve estar contida no método **main** e está diretamente vinculada (intrínseca) ao método **start**.

O método **start** é o responsável por iniciar a apresentação da **Application JavaFX**; portanto, ele deve conter as instruções essenciais para colocar uma aplicação na tela do dispositivo do usuário, como, por exemplo, em um televisor de um telespectador ou em um dispositivo móvel. A classe **Stage** é a responsável por construir um plano de visualização. Uma **Application JavaFX** pode ter vários planos simultâneos de visualização e existe opção selecionar aquele que deve ou não estar visível. A aplicação JavaFX define a interface do usuário por meio de um objeto da classe **Stage** e de um objeto da classe **Scene**. A classe **Scene** é o recipiente para todo o conteúdo. Exemplo 1-1 cria o palco e cena e faz a cena visível em um tamanho determinado pixel. Conforme exemplo, a **figura 1** apresenta o código-fonte suficiente para criar uma cena gráfica e inserir um objeto gráfico (*widget*) nessa cena; dá-se o nome de objeto gráfico, quando o *widget*

não desempenha um papel ativo na cena, e dá-se o nome de ator, quando o *widget* desempenha uma atuação na cena, como, por exemplo, participa de uma animação; a **figura 2** ilustra a interface gráfica da aplicação durante a execução, apresentando um widget que se comporta como objeto gráfico.

Example 1-1 Hello World

```
package helloworld;

import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.layout.StackPane;
import javafx.stage.Stage;

public class HellowWorld extends Application {
    public static void main(String[] args) {
        launch(args);
    }

    @Override
    public void start(Stage primaryStage) {
        primaryStage.setTitle("Hello World!");
        Button btn = new Button();
        btn.setText("Say 'Hello World'");
        btn.setOnAction(new EventHandler<ActionEvent>() {

            @Override
            public void handle(ActionEvent event) {
                System.out.println("Hello World!");
            }
        });

        StackPane root = new StackPane();
        root.getChildren().add(btn);
        primaryStage.setScene(new Scene(root, 300, 250));
        primaryStage.show();
    }
}
```

Figura 1: Código-fonte de Referência - “Hello World!” (Oracle, 2012)



Figura 2: Estilo Básico de Interface JavaFX - “Hello World!” (Oracle, 2012)

4 Arquitetura e Framework JavaFX

Conforme citado na **seção 1**, JavaFX é uma plataforma para construção de aplicações ricas baseadas em tecnologia Java. A **figura 3** ilustra os componentes da arquitetura JavaFX; nas seções seguintes, apresenta-se uma breve descrição de cada um desses componentes, permitindo melhor compreensão do *framework* como um todo.

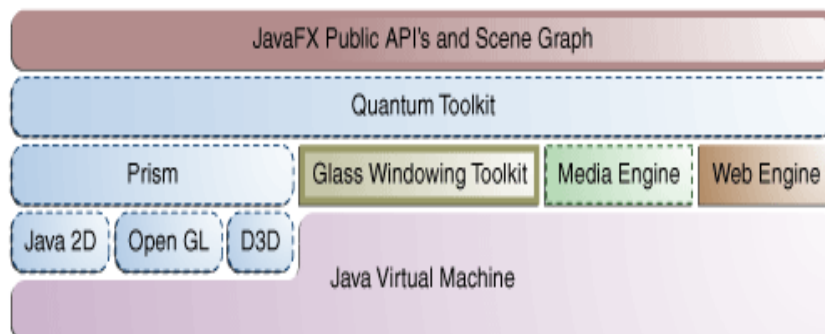


Figura 3: Arquitetura JavaFX (Oracle, 2012)

4.1 Cena Gráfica ou Representação Gráfica da Cena

Uma cena gráfica JavaFX é o ponto inicial para a construção de uma aplicação JavaFX. De forma conceitual é uma árvore hierárquica de nós (ou **nodes**), que representam todos os elementos visuais da interface com o usuário, podendo conter entradas ou objetos renderizados. A biblioteca **javafx.scene** permite a criação e especificação de vários tipos de conteúdo, tais como:

- nós (**nodes**): formas (2D e 3D), imagens, mídia, navegador web embutido, texto, controles da

interface do usuário, gráficos, grupos e *containers*;

- estado (**state**): transformações (posicionamento e orientação dos nós), efeitos visuais e outros estados visuais do conteúdo gráfico e/ou multimídia;
- animações (**animation**): variação/mutação das propriedades de objetos da cena ao longo do tempo;
- efeitos (**effects**): objetos simples que mudam a aparência dos nós da cena, tais como borrões, sombras e ajustes de cor/tonalidade.

4.2 Sistema Gráfico

O sistema gráfico da plataforma JavaFX, mostrado em azul na arquitetura apresentada na **figura 3**, é um visto como um *middleware* implementado entre a camada da cena gráfica JavaFX e uma máquina virtual Java (JVM). Tal implementação suporta tanto cenas 2D quanto 3D, fornecendo as camadas de software necessárias aos processos de renderização, quando o hardware gráfico do sistema é insuficiente para suportar renderização acelerada. O componente *Prism* gerencia todas as tarefas de renderização, podendo ser configurado para renderização tanto em hardware quanto em software, incluindo 3D (Oracle, 2012).

4.3 Glass Windowing Toolkit

O subsistema Glass Windowing Toolkit (GTK), apresentado em tons de bege na arquitetura apresentada na **figura 3**, é o nível mais baixo da pilha gráfica do *framework* JavaFX. Sua principal tarefa é fornecer serviços nativos do sistema operacional, tais como o gerenciamento de janelas, *timers* e superfícies (*surfaces*), servindo como uma camada independente de plataforma, que conecta a plataforma JavaFX ao sistema operacional da máquina (Windows, Linux, Android, dentre outros). O GTK é também responsável por gerenciar filas de eventos (Oracle, 2012).

4.4 Navegador Web Embutido

Uma das principais características da plataforma JavaFX é a implementação nativa de um navegador web, que possa ser embutido como objeto gráfico em uma aplicação JavaFX. Trata-se de um novo componente de interface com o usuário que fornece um visualizador web (baseado no webkit – um navegador opensource, que suporta HTML5, CSS, JavaScript, DOM e SVG) e todas as funcionalidades de navegação através de uma API. Um classe **WebView** encapsula um objeto da classe **WebEngine**, incorporando conteúdo HTML em cenas das aplicações, fornecendo campos e métodos para aplicar efeitos e transformações.

Referências Bibliográficas

(Oracle, 2012)

What Is JavaFX? Disponível em <http://docs.oracle.com/javafx/2/architecture/jfxpub-architecture.htm>. Visitado em junho de 2012.

(Santos Junior, 2007)

SANTOS JUNIOR, João Benedito dos et al. Prototyping Interactive Digital Television Programs Using Java and XML: A Platform for Developers and Viewers. Proceedings of EuroITV. EuroITV2007, Amsterdam, The Netherlands, 2007.

(Santos Junior, 2012)

Santos Junior, João Benedito dos. The JavaFX Platform for TV Profile. Pontifícia Universidade Católica de Minas Gerais, Laboratório de Televisão Digital Interativa. Poços de Caldas. 2012. Disponível em <http://tvdilab.inf.pucpcaldas.br>