Virtual Everywhere
January 24, 2023

Say the Words: Modern Java with JavaFX for Rich Client UIs

Paul Anderson
Gail Anderson
Anderson Software Group, Inc.
asgteach.com

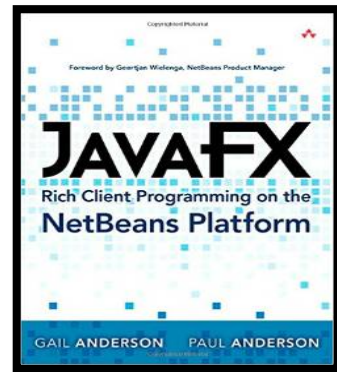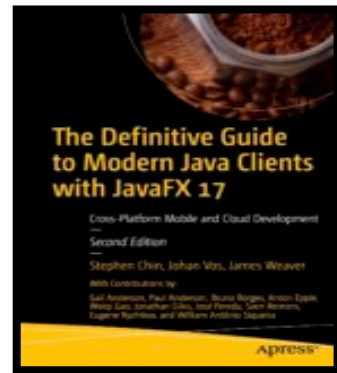# So Who Are We?

▶ Training Company
  ◦ Java, JavaFX Courses

▶ Books
  ◦ Definitive Guide to Modern Java Clients with JavaFX 17
  ◦ JavaFX Rich Client Programming on the NetBeans Platform

▶ Affiliations
 ◦ Java Champions
  ◦ Oracle Ace Members

Agenda

- Why JavaFX on Mobile?
- Gluon Substrate and GraalVM
- GluonFX Plugin for Maven
- Mobile App Examples
  - TiltMaze – The Accelerometer
 - Wordish – FXML and Scene Builder
- JavaFX Techniques
- Summary, Q & A

Why JavaFX on Mobile?

- Critical Goal
  - Platform independent source code
  - "Write Once, Install Everywhere"
- Design Approach
  - Gluon tools and GraalVM platform
  - Lets you build native images
- JavaFX Advantages
  - Java UI, scene graph, nodes, FXML views, charts
  - Properties, listeners, binding, event handlers
 - JavaFX is Java, so you get a modern language

Gluon Substrate and GraalVM
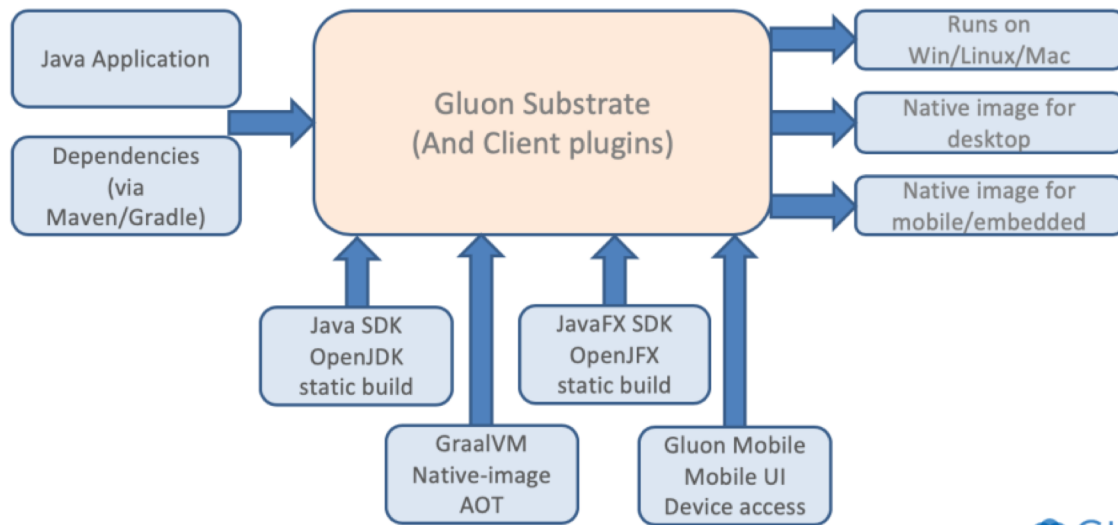
TiltMaze
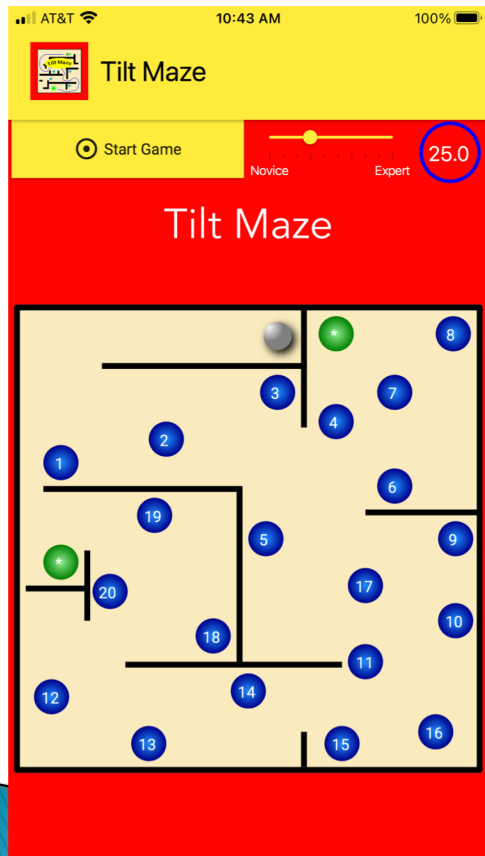
- [TiltMaze on Google Play](#)
- [TiltMaze on Apple App Store](#)
- Features
  - Access device accelerometer in Animation loop
  - Gluon Mobile for mobile-friendly controls (licensed)
  - FXML to keep View and Controller separate
  - Collision detection to control game
  - Property binding to keep UI in sync

Wordish
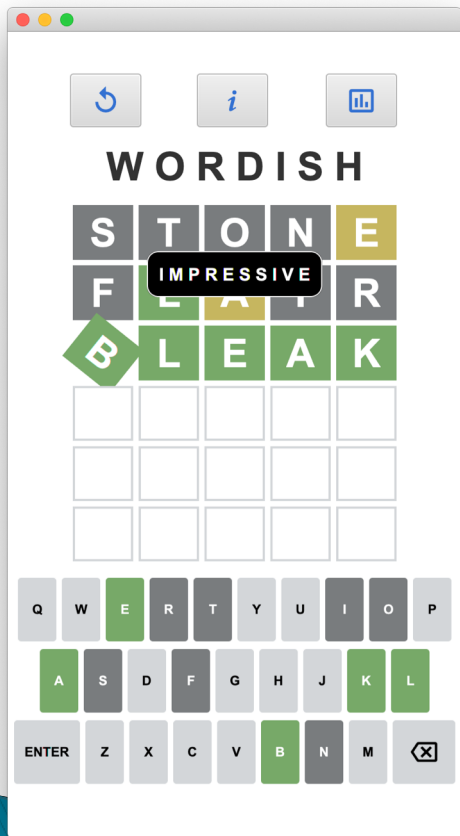
▶ Kind of like Wordle

▶ Features

◦ Advanced UI layout concepts

◦ Use Scene Builder/FXML to design UI

◦ CSS Pseudo classes

◦ Third-party Font icons

◦ Node-level animation

◦ Leverage modern Java

◦ Property binding to keep UI in sync

◦ Not in app stores, on github: https://github.com/gailasgteach/Wordish

GluonFX plugin for Maven

▸ Starter Page + GraalVM download
  ◦ https://start.gluon.io/
  ◦ https://github.com/gluonhq/graal/releases/tag/gluon-22.1.0.1-Final
▸ Samples on github
  ◦ https://github.com/gluonhq/gluon-samples
  ◦ Starter projects you can download and study
▸ Tutorial from Idea to App Store
  ◦ https://foojay.io/today/creating-mobile-apps-with-javafx-part-1/

Wordish – FXML and Scene Builder

▶ Scene Builder
  ◦ Generates FXML
  ◦ Use drag and drop to design view
  ◦ Customize for extended controls and Font Library

▶ Layout Controls
  ◦ GridPane for Button controls and title
  ◦ TilePane for letter Labels
  ◦ FlowPane for virtual keyboard

JavaFX Property Binding

- Property Binding
  - Properties depend on values of other properties
  - Succinct, less error prone, consistent game state
  - Avoid writing listeners
  - Build logical expressions with support APIs

    `.lessThan(), .or(), .isEqualto()`

- Wordish

JavaFX Pseudo Classes

- CSS Support for state of a JavaFX control
  - Used throughout JavaFX control hierarchy
  - Example: Button turns less opaque when disabled
- Wordish
  - LetterLabel – property LetterDisplay
  - KeyButton – property LetterDisplay
  - style.css – define styling for various states

JavaFX Animation in Wordish

▸ Node animation with Transitions
  ◦ TranslateTransition – move node
  ◦ FadeTransition – change opacity of node
  ◦ RotateTransition – rotate node
  ◦ ParallelTransition – transitions in parallel
  ◦ SequentialTransition – transitions sequentially
▸ Wordish

Chart Your Guesses

- JavaFX Charts
  - Adding data to the Bar Chart
  - Customizing the Bar Chart
  - Adding controls to a Chart Node
    - Create delay
    - Use snapshot
- Wordish

Summary

▸ JavaFX on Mobile
  ◦ Java and JavaFX code base for development
  ◦ TiltMaze and Wordish Apps
  ◦ GraalVM for native images on IOS/Android
▸ JavaFX Techniques
  ◦ FXML and Scene Builder
  ◦ Property Bindings
  ◦ Pseudo Classes for CSS Styling
  ◦ Animation, Charts

Q & A

- ## Thanks for Attending!

  paul.anderson@asgteach.com
  @paul_asgteach
  gail.anderson@asgteach.com
  @gail_asgteach
  asgteach.com

- ## GitHub Source Code
  https://github.com/gailasgteach/Wordish/
  https://github.com/gailasgteach/TiltMaze/