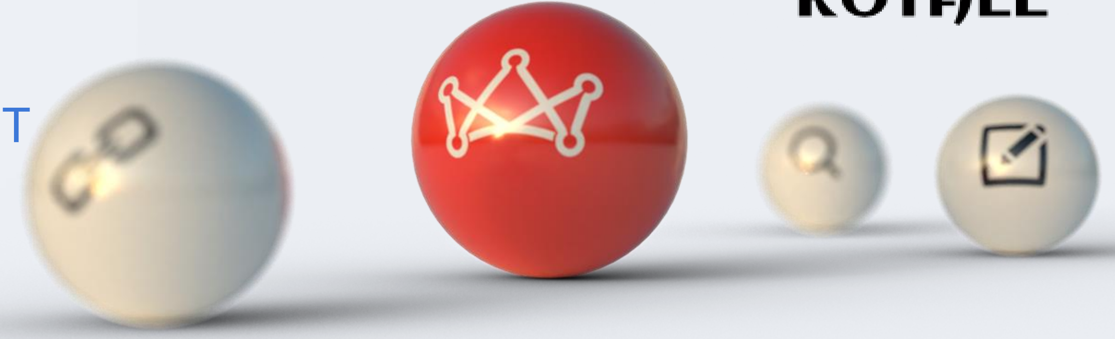
Apache Royale

Hello, Royale!

Apache ROYALE™



29.09.2020 18:55 CEST

Andrew Wetmore

Your presenter

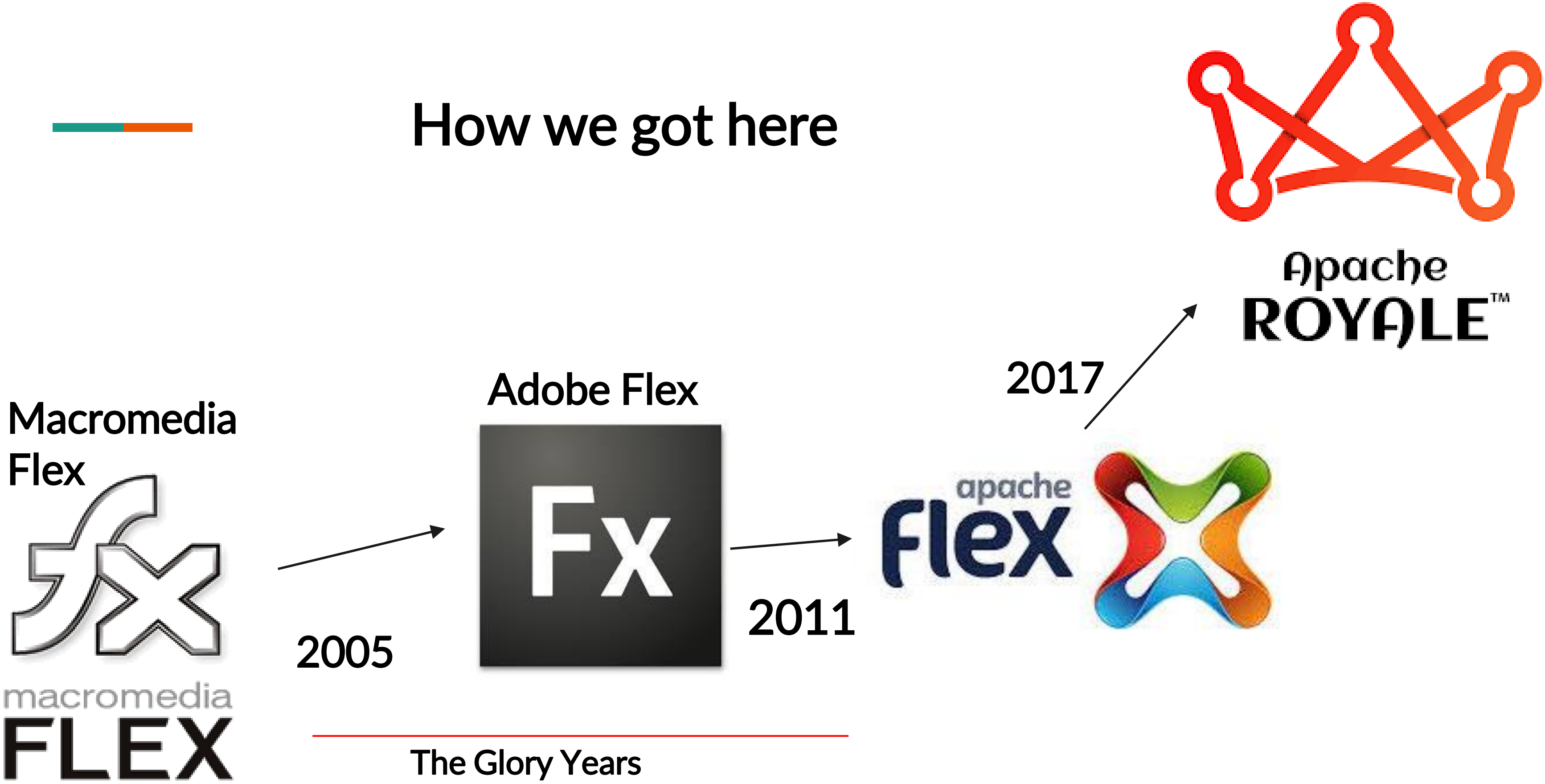
Andrew Wetmore - Apache Royale committer working on documentation; editor/writer for the Apache Infrastructure team.

andreww@apache.org

Apache Royale

- An SDK for developing data-driven applications
- From one code base, export to run
  - On any browser using JavaScript
  - On the desktop, using the Adobe AIR runtime or similar
  - On mobile devices through Apache Cordova
- Extensible to WASM, native iOS, Android; new technologies as they appear

How we got here



Macromedia Flex

2005

Adobe Flex

2011

2017

The Glory Years

Under the hood

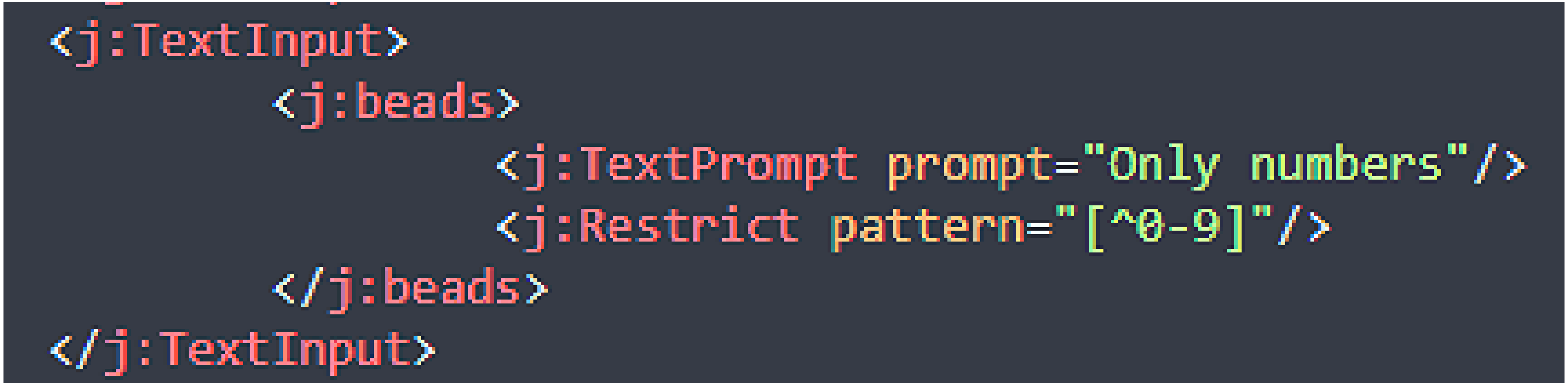
- **MXML declarative language** for defining and powering the user interface. Makes possible many of Royale's cool features
- **ActionScript** for data manipulation, graphics, functions that support the front end
- **AMF (ActionScript Message Format)** - powerful tool for remote object calls using JSON, and for local storage
- **Frameworks** like **Crux** to help develop efficient web applications - deploy Inversion of Control (IOC), dependency injection...

Concepts

- **OOP -** break down your software into bite-sized problems to create one object at a time.
  - **Modular** design
  - **Strictly-typed properties**

Concepts

- PAYG
  - more composition, less inheritance
  - Smaller footprint
  - Strands and beads



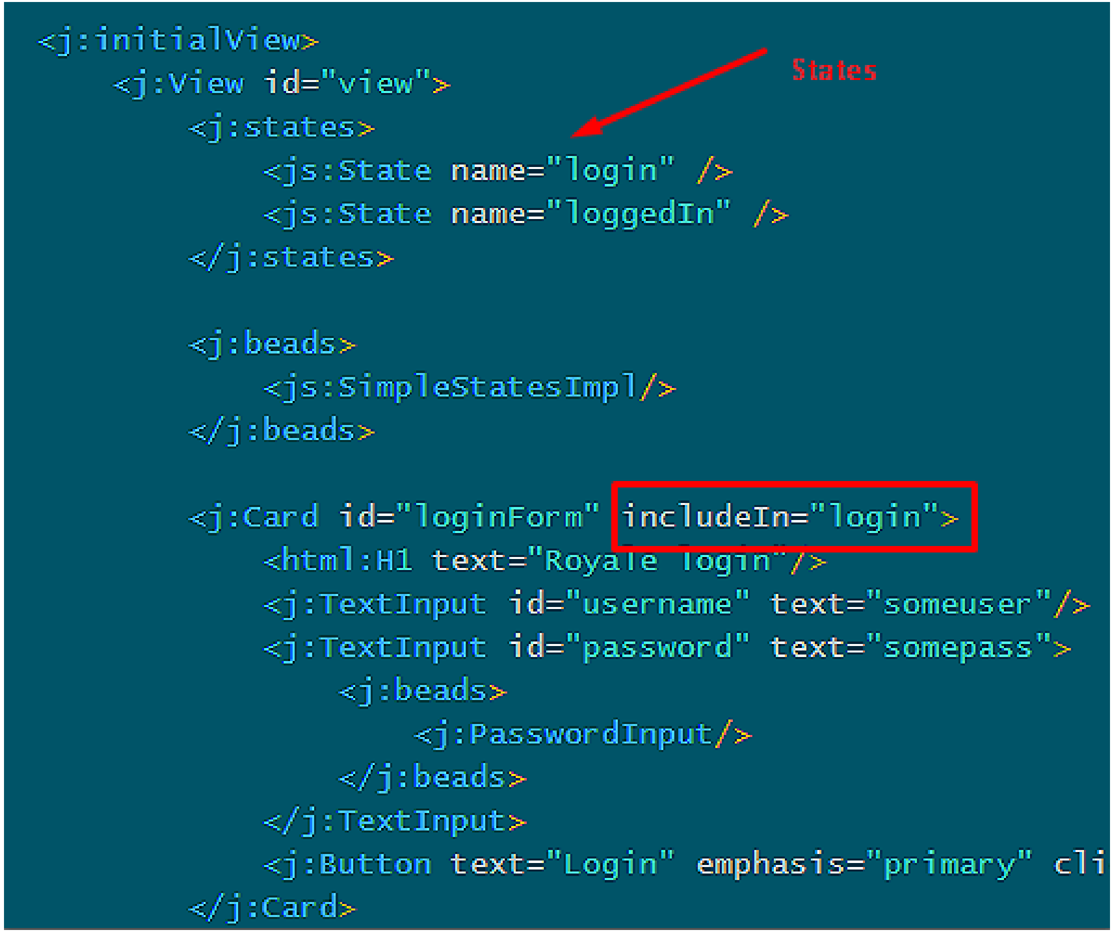
```
<j:TextInput>
        <j:beads>
                <j:TextPrompt prompt="Only numbers"/>
                <j:Restrict pattern="[^0-9]"/>
        </j:beads>
</j:TextInput>
```

Cool Features

- Data binding
- Event Bubbling
- Component Sets*
- ViewStates
- ItemRenderers
- Use external libraries

*Carlos Rovira presenting **Thursday** on the **Apache Jewel component set**

ViewStates

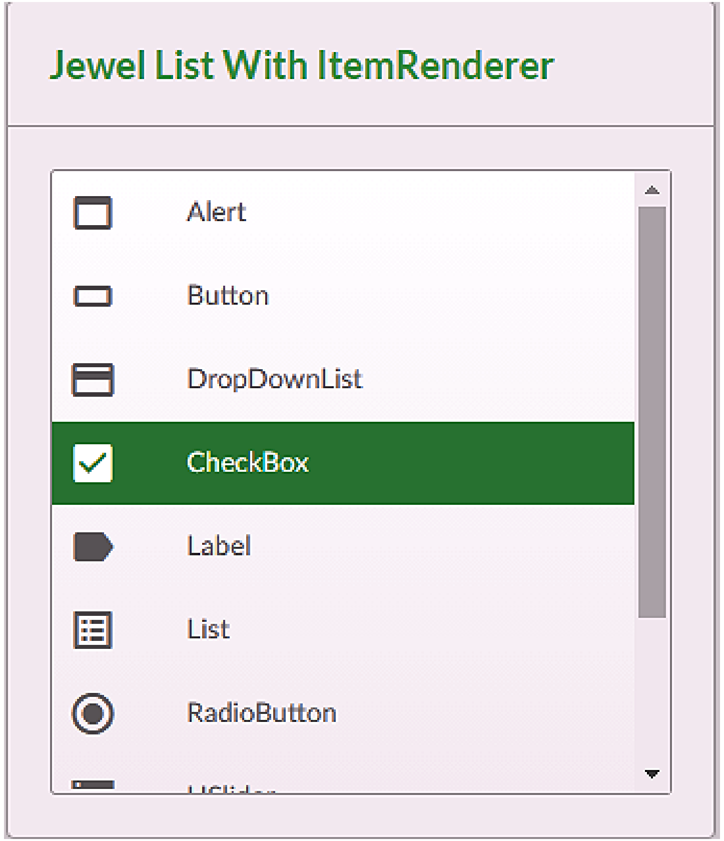


```
<j:initialView>
    <j:View id="view">
        <j:states>
            <js:State name="login" />
            <js:State name="loggedIn" />
        </j:states>

        <j:beads>
            <js:SimpleStatesImpl/>
        </j:beads>

        <j:Card id="loginForm" includeIn="login">
            <html:H1 text="Royale login"/>
            <j:TextInput id="username" text="someuser"/>
            <j:TextInput id="password" text="somepass">
                <j:beads>
                    <j:PasswordInput/>
                </j:beads>
            </j:TextInput>
            <j:Button text="Login" emphasis="primary" cli
        </j:Card>
```

ItemRenderers



Jewel List With ItemRenderer

- Alert
- Button
- DropDownList
- ✓ CheckBox
- Label
- List
- RadioButton

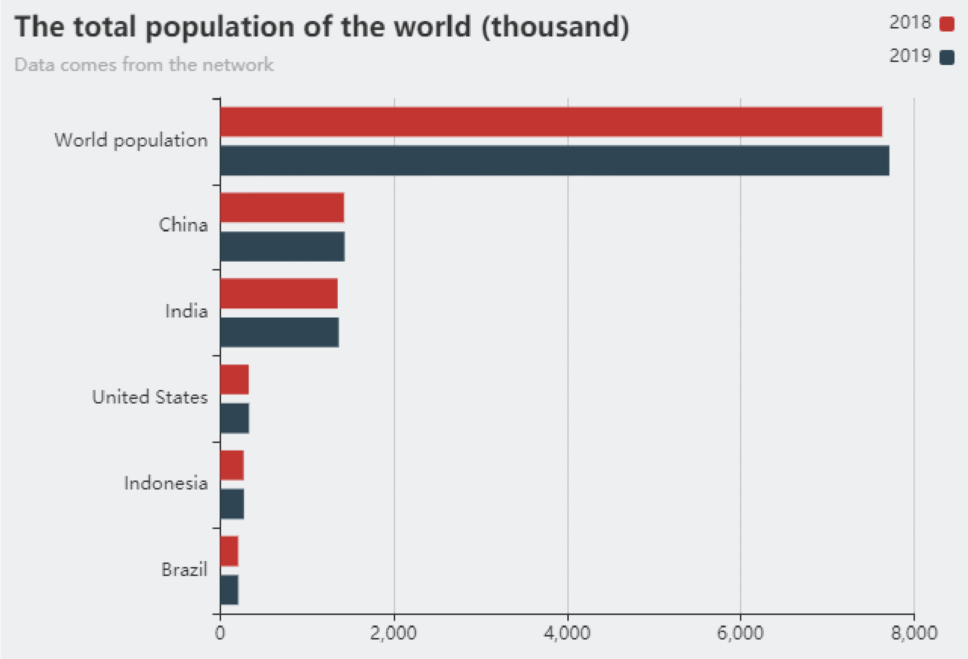
External library: ECHARTS

Apache eCharts is an incubating project. Hierarchical JSON format to declare components, styles, data, and interactions.

Five minutes' work and twenty lines of code to add eCharts to a Royale app, using either

- **@Externs** compiler directive
- **mx.external.ExternalInterface**, Royale's implementation of the Flash ExternalInterface

External library: ECHARTS


The total population of the world (thousand)

Data comes from the network

Example
from **María
Jose Estéve**

Royale is for Flex folks

Dealing with the End Of Flash

- **Move to some other technology**
  - Learn new technology
  - Rebuild everything

- **Migrate to Royale***
  - Gentle Flex => Royale learning curve
  - Reuse a ton of what you already have
  - Leverage existing skills
  - Much smaller footprint
  - Much faster data loading and display -

*Alina Kazi presenting **Wednesday: Moving an app from Flex to Royale**

Royale is for everyone else, too

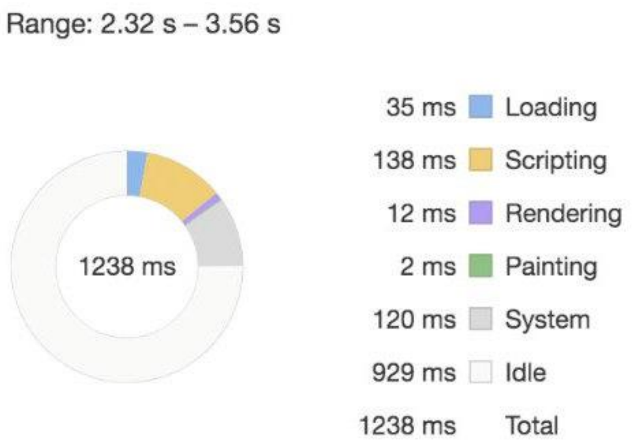
- (ymmv) **strongly typed properties**
- **OOP**
- **Frameworks** and **component sets**
- Incorporate the universe of **existing JS libraries**

Carlos Rovira presenting **Thursday: Starting from a blank file**

Oh, and it's very fast

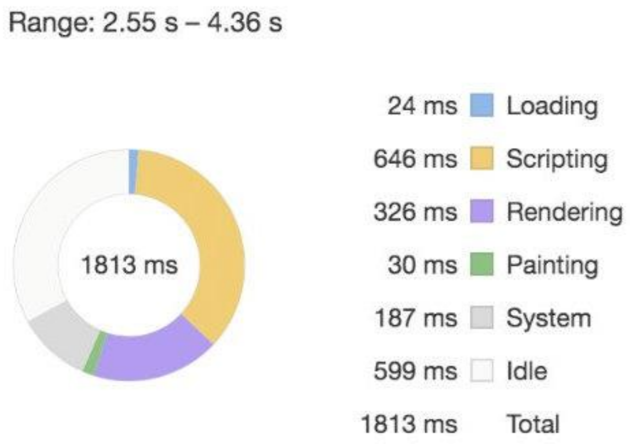
Royale
VS
ReactJS

DevTools - unhurdle.github.io/spectrum-royale/


Range: 2.32 s – 3.56 s

35 ms	🟦	Loading
138 ms	🟨	Scripting
12 ms	🟪	Rendering
2 ms	🟩	Painting
120 ms	⬜	System
929 ms	◻	Idle
1238 ms		Total

DevTools - opensource.adobe.com/spectrum-css/components/accordion/


Range: 2.55 s – 4.36 s

24 ms	🟦	Loading
646 ms	🟨	Scripting
326 ms	🟪	Rendering
30 ms	🟩	Painting
187 ms	⬜	System
599 ms	◻	Idle
1813 ms		Total

Summing up:

Code once, run anywhere

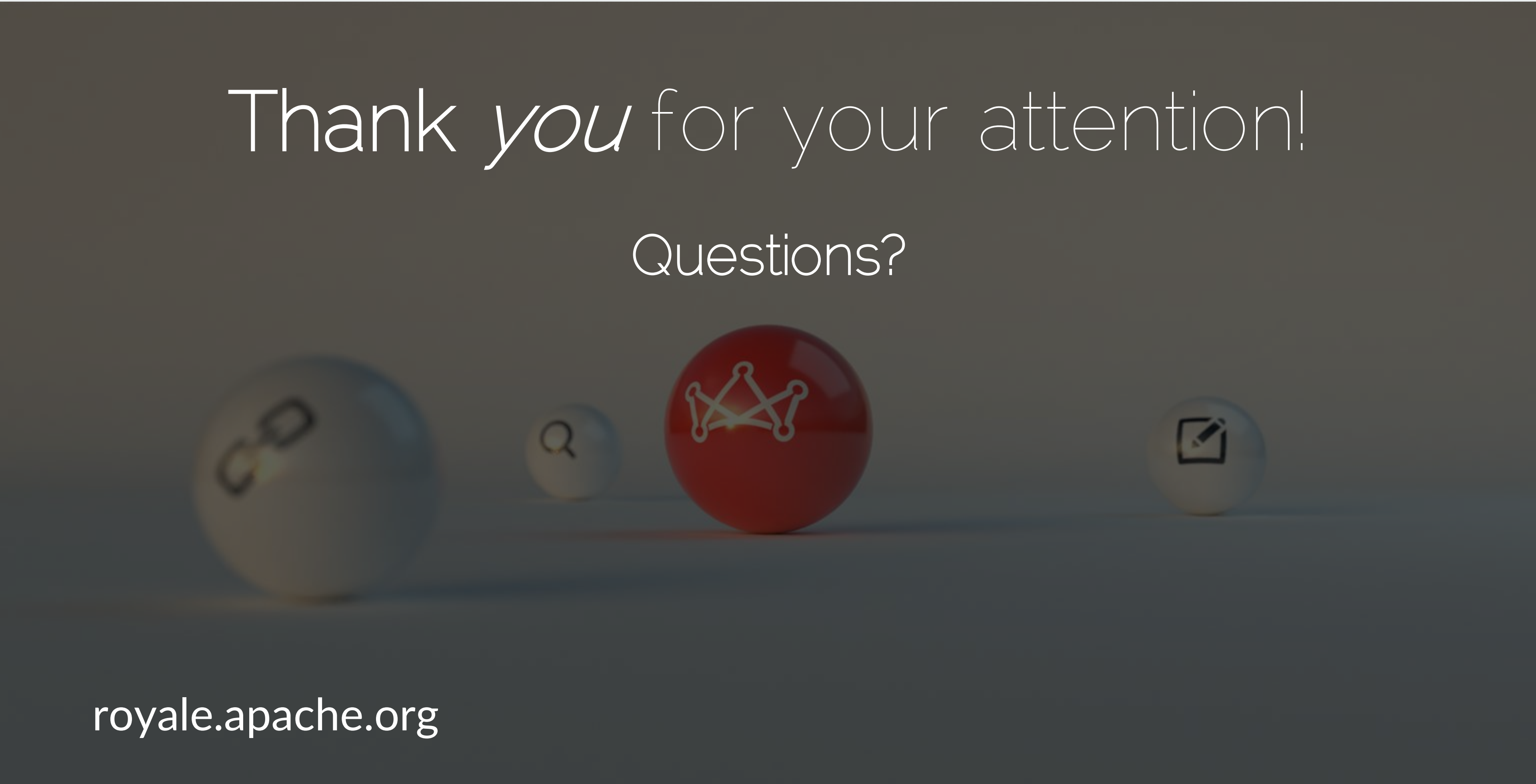
Code for the future

Open source! No "Flash extinction event" ahead of us

Apache
ROYALE™

Thank *you* for your attention!

Questions?



royale.apache.org