



Titanium Mobile: Best Practices for Android

Josh Roesslein, Software Engineer

Appcelerator, Inc.

jroesslein@appcelerator.com

Max Stepanov, Senior Software Engineer

Appcelerator, Inc.

mstepanov@appcelerator.com

Agenda

- History
- Windows and Navigation
- Layout events
- Event Bubbling
- Modern Look and Feel
- Debugging

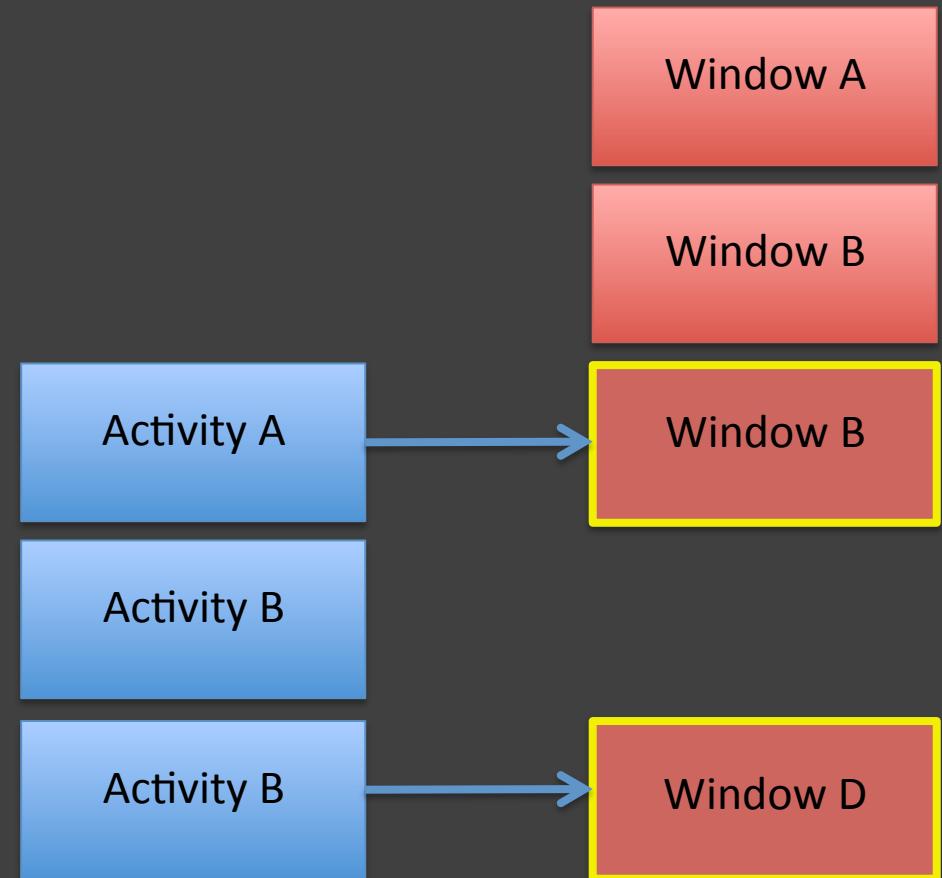
History

- 0.8: HTML UI / WebView
- 0.9: Native UI / Mozilla Rhino
- 1.8: Google V8

Windows and Navigation

- Heavyweight vs. Lightweight
- Activities and Back Stack
- Back Stack

Windows and Navigation



Layouts

```
1 var win = Ti.UI.createWindow({});  
2  
3 var view = Ti.UI.createView({  
4     height: 100,  
5     width: 50  
6});  
7  
8 win.addEventListener('open', function() {  
9     // This could be 0  
10    Ti.API.info('open: ' + view.rect.height);  
11});  
12  
13 win.addEventListener('postlayout', function() {  
14     // This will return 100  
15     Ti.API.info('postlayout: ' + view.rect.height);  
16});  
17  
18 win.add(view);  
19 win.open();  
20 |
```

- Eventing: Use *postlayout* NOT *open*
- A layout pass may not have occurred by the time an *open* event is fired
- A *postlayout* event occurs after a native Android layout pass (may occur multiple times)

Form Factor



- Targeted set of devices
- Multiple sets of assets
- Resources/assets
- Densities

Manifest/Themes

- Generated Manifest
- Default Theme
- Custom Manifest
- tiApp.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <!-- A simple theme based on the
        default theme for Android API
        level 10 and below -->
    <style name="Theme.MyTheme"
          parent="android:Theme">
        <item
            name="android:windowBackground">
            @drawable/background
        </item>
    </style>
</resources>
```

Manifest/Themes

- Generated Manifest
- Default Theme
- Custom Manifest
- tiApp.xml

Theme: platform/android/res/values

```
<android xmlns:android="http://schemas.android.com/apk/res/android">
    <tool-api-level>14</tool-api-level>
    <manifest>
        <uses-sdk
            targetSdkVersion="14"/>
    </manifest>
</android>
```

Event bubbling

- SDK 3.0 allows control over event bubbling
- Titanium.Event object has two new properties
 - *bubbles*
 - *cancelBubble*
- Current defaults are identical to 2.X, but will change to achieve parity in the future

Control bubbling

Allow custom events to bubble

```
button.fireEvent("myEvent",  
    {bubbles: true});
```

Allow bubbling to cancel during event handling

```
view.addEventListener("myEvent",  
    function(e) {  
        e.cancelBubble = true;  
    })
```

Cancel bubbling

Allow views to bubble up / not to bubble up events

```
view.bubbleParent = false;
```

For more information:

http://docs.appcelerator.com/titanium/3.0/index.html#!/guide/Event_Handling

Newer Look/Feel

- Target API level
 - Backwards compatible
 - Unlocks newer behaviors and features
- Themes
 - Consistent
 - Holo themes
- ActionBar
 - MenuItem
 - Action Items
 - Tabs
 - Opt-in
 - TabGroup API

tiapp.xml

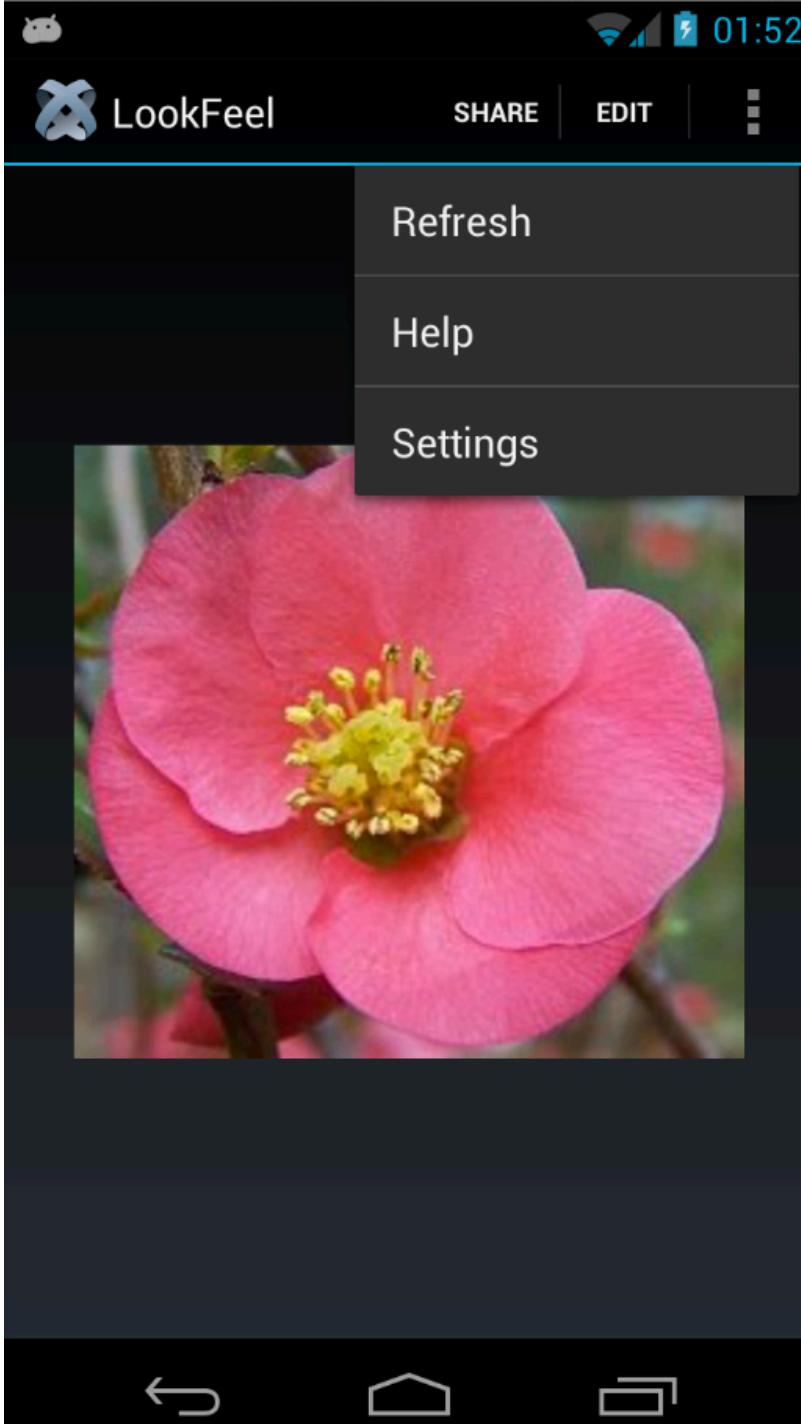
```
<android xmlns:android="...">
  <manifest>
    <uses-sdk android:minSdkVersion="8"
              android:targetSdkVersion="16"/>
  </manifest>
</android>
```

platform/android/res/values-v11/theme.xml

```
<resources>
  <style name="Theme.Titanium"
        parent="android:Theme.Holo.Light"
        />
</resources>
```

New MenuItem Properties for ActionBar

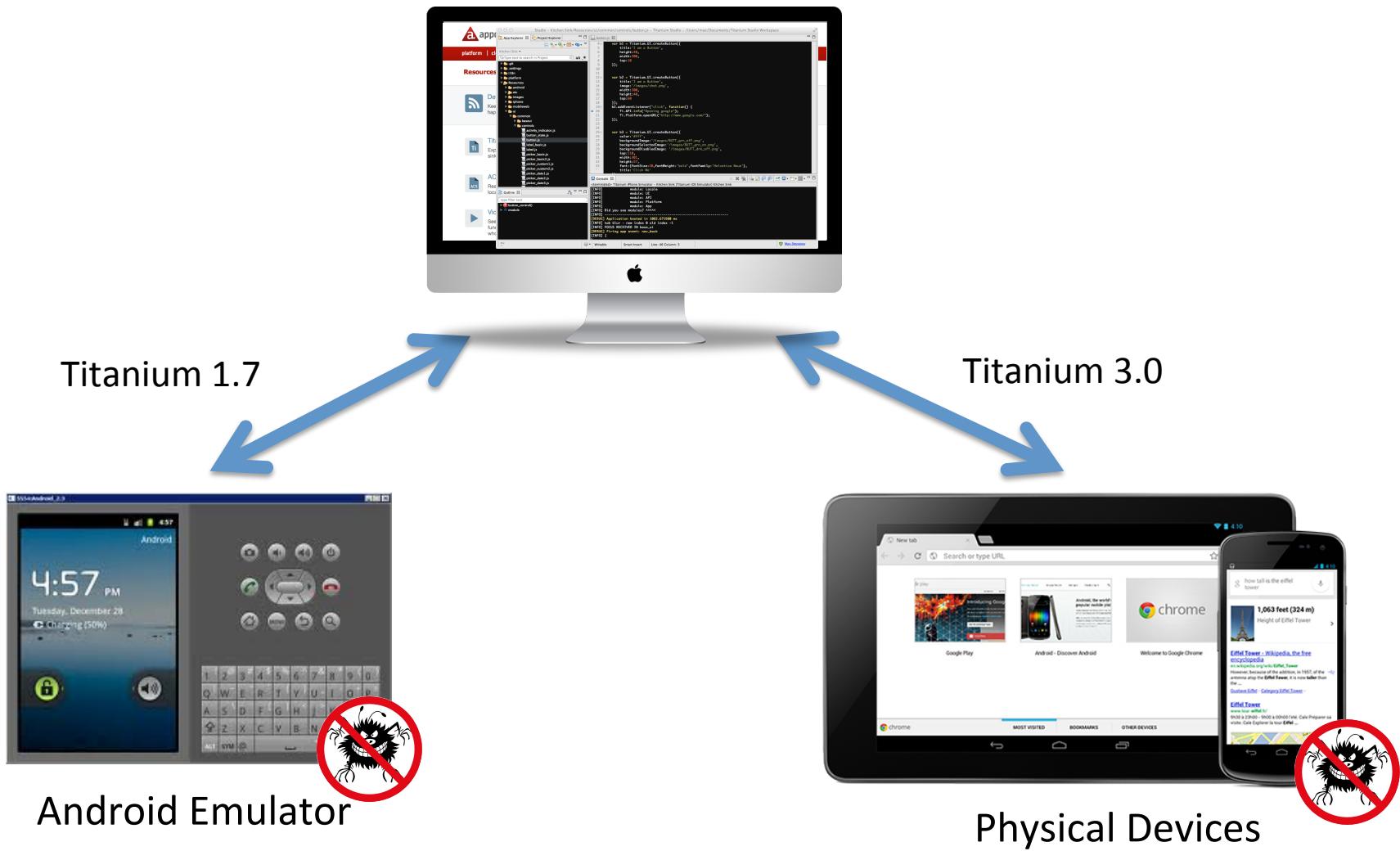
```
win.activity.onCreateOptionsMenu = function(e) {
  var item = e.menu.add({ title: "Share" });
  item.showAsAction =
    Ti.Android.SHOW_AS_ACTION_ALWAYS;
  // etc.
};
```



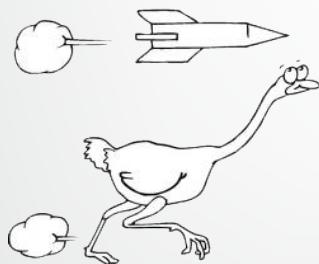
Android Debugging



New in Titanium



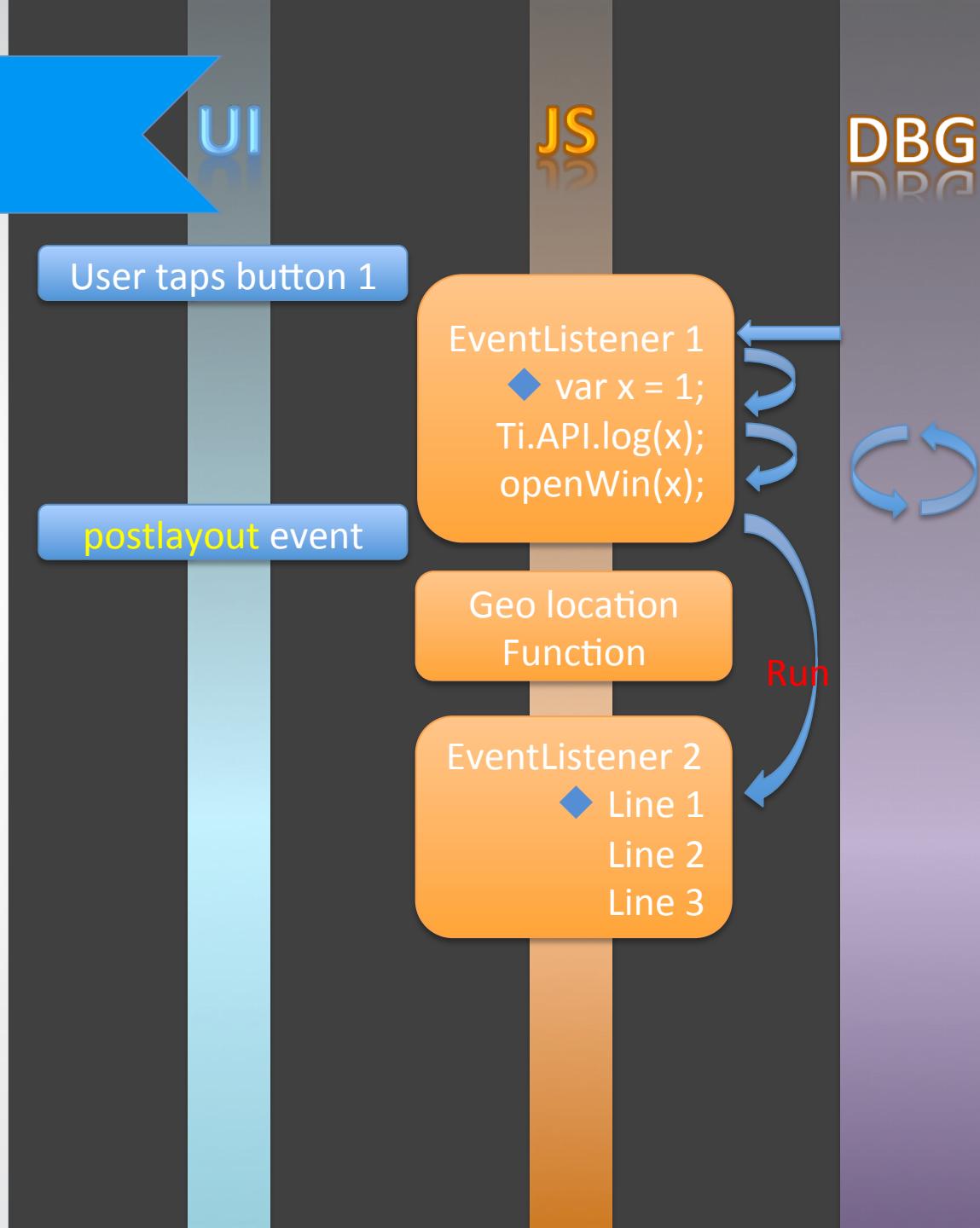
Device Debugging



- V8 only (for on-device debugging)
- Just plug in device into USB
- Launches faster than Emulator!

Threading

- UI thread for handling user interactions
- JS thread for the application logic
- Debugger thread for communications with Titanium Studio
- Other Android platform threads



Best Practices



- Use conditional breakpoints
 - JavaScript code block
 - Hit count
- Use Console logging with Ti.API functions

Q&A



Josh Roesslein & Max Stepanov

jroesslein@appcelerator.com / mstepanov@appcelerator.com