



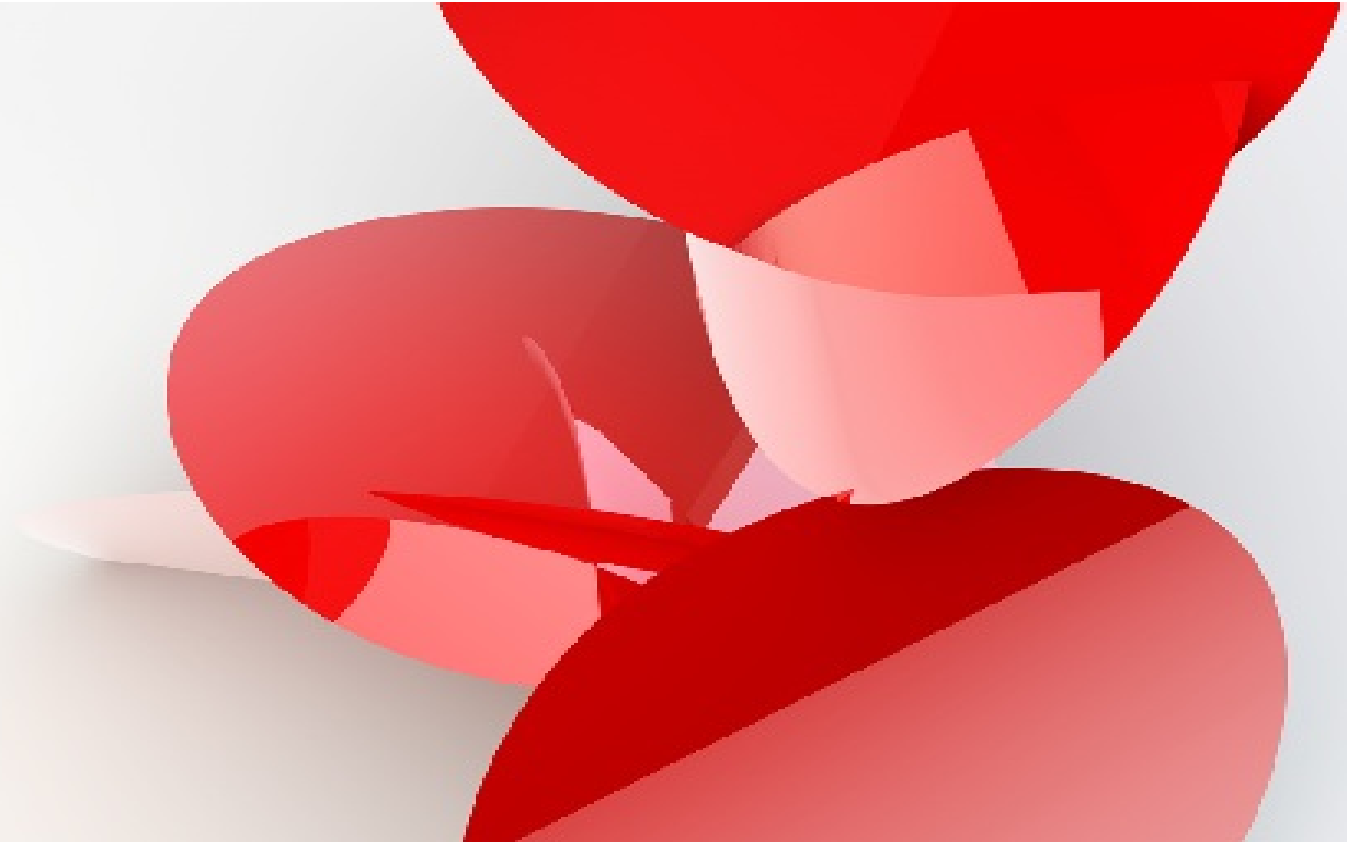
Christophe Coenraets | Adobe

Building Enterprise Mobile Applications with Flex and Java



Building Enterprise Mobile Applications with Flex

Christophe Coenraets – <http://coenraets.org> – @ccoenraets



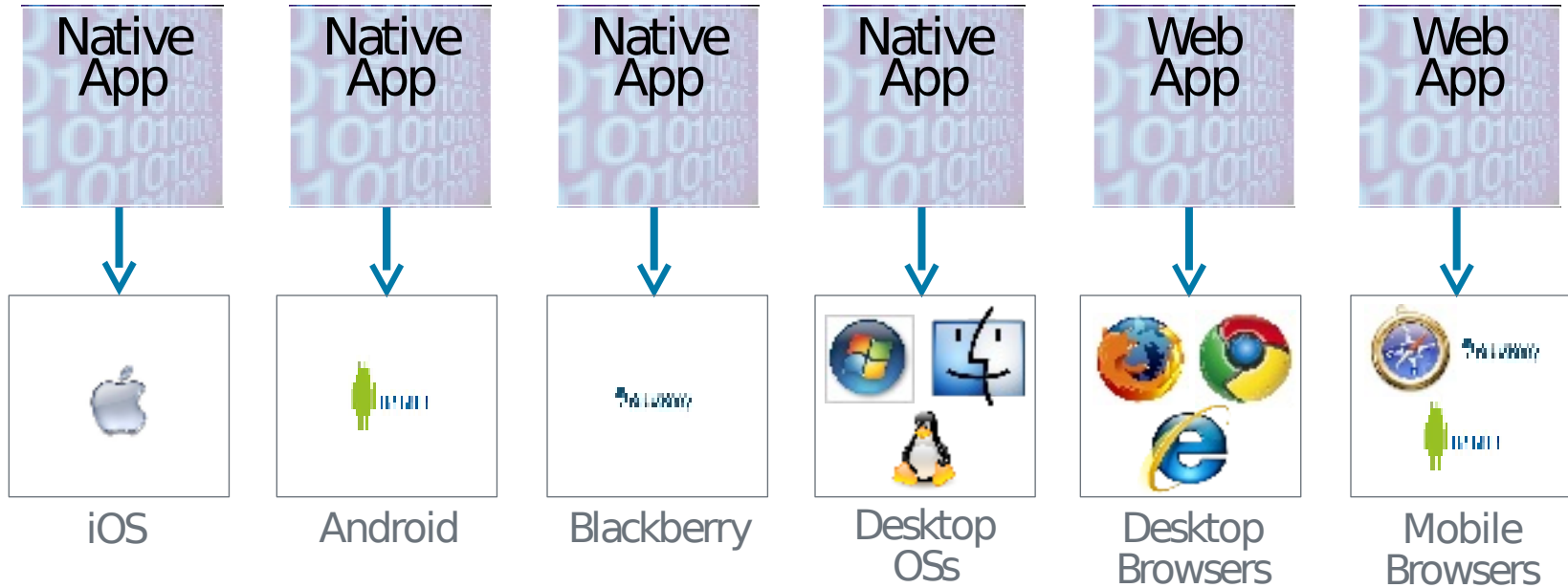
The explosion of devices introduces new challenges for developers



Challenges

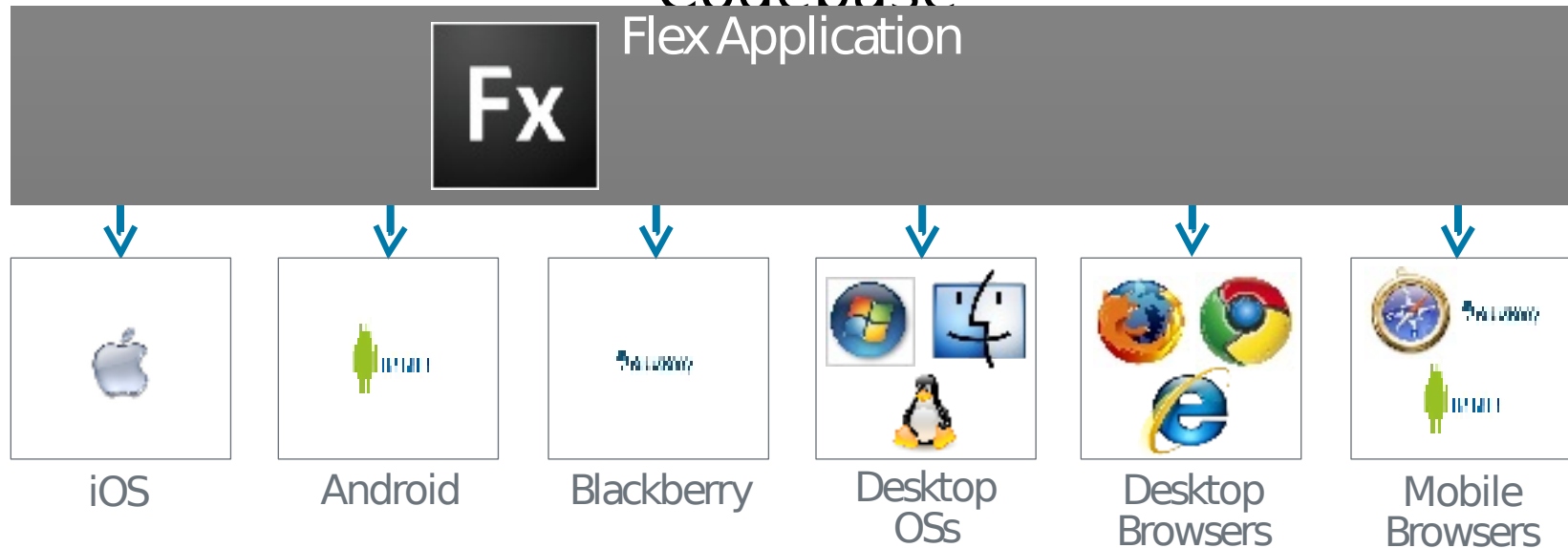
- Many Platforms
- Many Screen Resolutions
- Many Screen Densities
- Consistent experience
- Tools to develop, debug, profile
- Data Integration

Native Approach



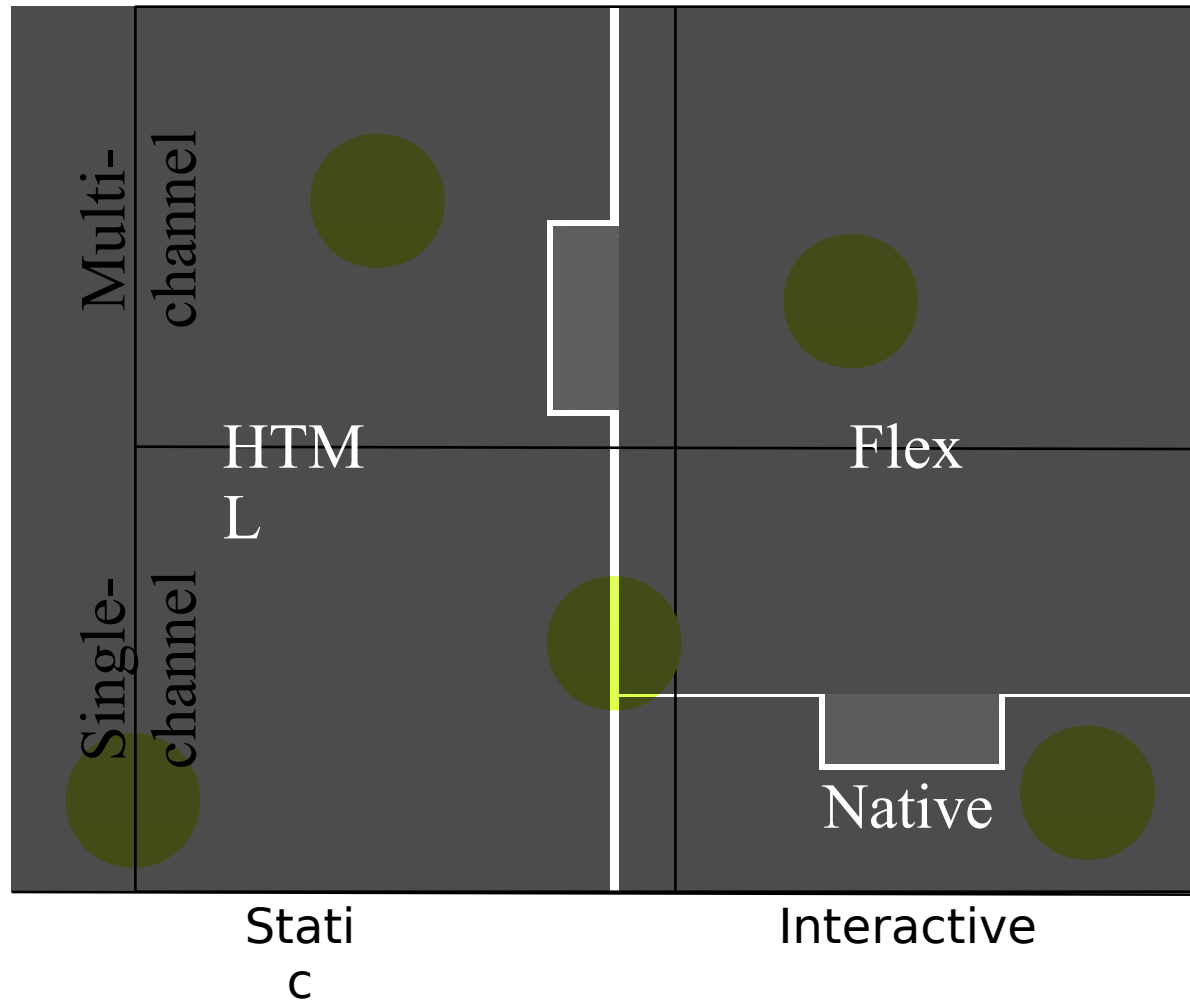
One Codebase Approach

One Tool, One Language, One
Codebase



Any Platform

Mobile Application Landscape



Flash Mobile Platform Benefits

- Consistency
- Expressiveness
- Performance
- Productivity
- Tools: Develop – Debug – Profile
- Designer/Developer Workflow

Demo



| | |
|------------------------|--------------------------|
| GPS | Remote Method Invocation |
| Accelerometer | Messaging / Push |
| Multi-Touch | Peer-to-Peer |
| Camera | VOIP |
| Native App Integration | Collaboration |
| Local Database | Video-Conferencing |
| Stage Web View | |

GPS

- `Geolocation.isSupported`
- `addEventListener(GeolocationEvent.UPDATE, callbackFunction);`
- `GeolocationEvent`:
 - `altitude`
 - `heading`
 - `horizontalAccuracy`
 - `latitude`
 - `longitude`
 - `speed`
 - `timestamp`
 - `verticalAccuracy`

Accelerometer APIs

- `Accelerometer.isSupported`
- `addEventListener(AccelerometerEvent.UPDATE, callbackFunction);`
- `AccelerometerEvent`:
 - `accelerationX`
 - `accelerationY`
 - `accelerationZ`
 - `timestamp`

Multitouch -- MultitouchInputMode.TOUCH_POINT

TouchEvent

- TOUCH_BEGIN
- TOUCH_END
- TOUCH_MOVE
- TOUCH_OUT
- TOUCH_OVER
- TOUCH_ROLL_OUT
- TOUCH_ROLL_OVER
- TOUCH_TAP

Gestures -- MultitouchInputMode.GESTURE

- TransformGestureEvent
 - GESTURE_PAN
 - GESTURE_ROTATE
 - GESTURE_SWIPE
 - GESTURE_ZOOM
- GestureEvent.GESTURE_TWO_FINGER_TAP
- PressAndTapGestureEvent.GESTURE_PRESS_AND_TAP

Camera Access

- CameraUI - Native Camera App
`var cameraUI:CameraUI = new CameraUI();`
`cameraUI.launch(MediaType.IMAGE);`
- CameraRoll - Choose photos from the device's camera roll
`var cameraRoll:CameraRoll = new CameraRoll();`
`cameraRoll.browseForImage();`
`cameraRoll.addBitmapData(bd);`

Native App Integration

- Open Email, Browser, Maps, Phone, or SMS
 - `navigateToURL(new URLRequest('mailto:foo@bar.com'));`
 - `navigateToURL(new URLRequest('http://www.coenraets.org'));`
 - `navigateToURL(new URLRequest('http://maps.google.com/'));`
 - `navigateToURL(new URLRequest('tel:1234567890'));`
 - `navigateToURL(new URLRequest('sms:1234567890'));`

Local DB API -- SQLite

```
var stmt:SQLStatement = new SQLStatement();  
stmt.sqlConnection = sqlConnection;  
stmt.text = "INSERT into foo values(:bar)";  
stmt.parameters[":bar"] = textInput.text;  
stmt.execute();
```

StageWebView

- **WebStageView.isSupported**

```
var swv:StageWebView = new StageWebView();
```

```
swv.viewPort = new Rectangle(0, stage.height - height, width, height);
```

```
swv.stage = stage;
```

```
swv.loadURL("http://www.coenraets.org");
```

Tour de Flex

Multiple densities: The problem

150 x 40 pixel
button



Desktop
monitor
@100 dpi
= 1.5" x 0.4"



Galaxy Tab
@160 dpi
= 0.9" x
0.25"



Droid 2
@240 dpi
= 0.6" x
0.17"

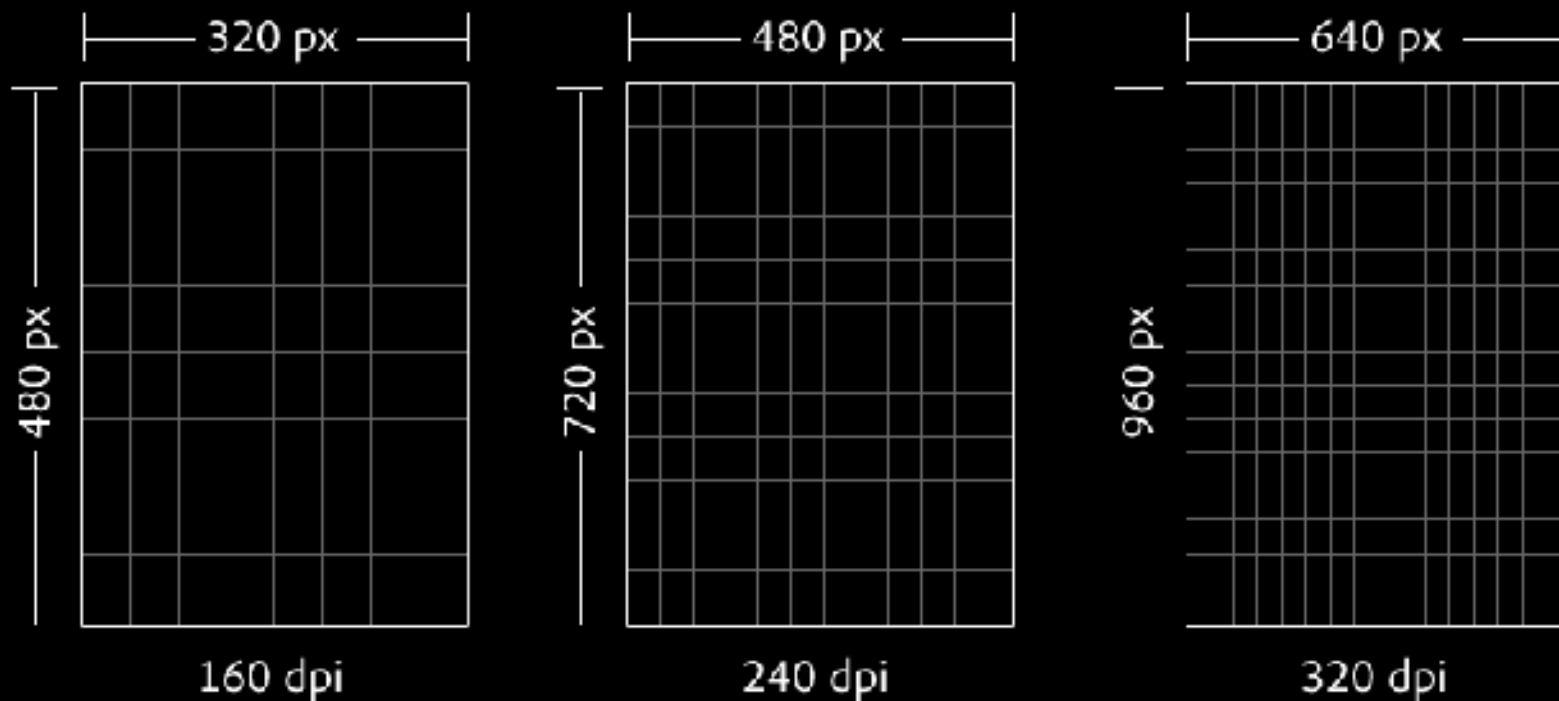


iPhone 4
@320 dpi
= 0.46" x
0.13"

Same pixel count, different
physical sizes
(Minimum recommended size: 0.25" x 0.25")

Another perspective

3.5" diagonal
screen



Same physical size, different pixel counts

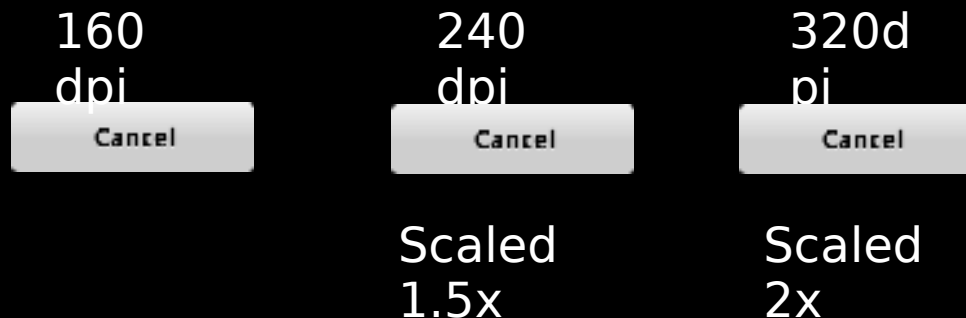
Default mapping for DPI classifications

Flex groups devices into **DPI classifications** based on actual device density

| Classification | 160 DPI | 240 DPI | 320 DPI |
|----------------------|--|---|--------------------|
| Devices | Most tablets iPhone 3GS Motorola Droid Pro | Most Android phones | iPhone 4 |
| Mapped range | < 200 DPI | >= 200 DPI <= 280 DPI | > 280 DPI |
| Typical Range | 132 DPI (iPad) to 181 DPI (HTC Hero) | 217 DPI (HTC Evo) to 254 DPI (NexusOne) | 326 DPI (iPhone 4) |

Solution 1: Automatic scaling for different DPIs

```
<Application applicationDPI="160">  
  <Button width="160" height="40"/>  
</Application>
```



Scaling different types of objects



Lorem

Ipsum

Dolor



Vectors
scale up well
(scaling down can be bad)
Outlines may blur slightly

Text
scales up well
(Flash scales font size)

Bitmaps
do not scale up well

Solution 2: Manual DPI management

- Leave applicationDPI unset (will default to same as runtimeDPI)
- Built-in component skins in mobile theme will adapt to different DPIs
- Your own layouts and skins will need to adapt (pixel and font sizes)
 - Can use @media to set CSS rules for different DPIs
 - Use data binding or code to adapt layout properties per DPI
 - Multi-DPI bitmaps still work

Flex density concepts: Multi-DPI bitmaps

```
<Button click="dealSummaryList.refresh()">
  <icon>
    <MultiDPIBitmapSource
      source160dpi="@Embed('assets/refresh160.png')"
      source240dpi="@Embed('assets/refresh240.png')"
      source320dpi="@Embed('assets/refresh320.png')"/>
    </icon>
  </Button>
```

UI differences across platforms

Android phone

No back button

Multiple actions



Title left-aligned

Flat-look buttons

iPhone

On-screen back button

Single action



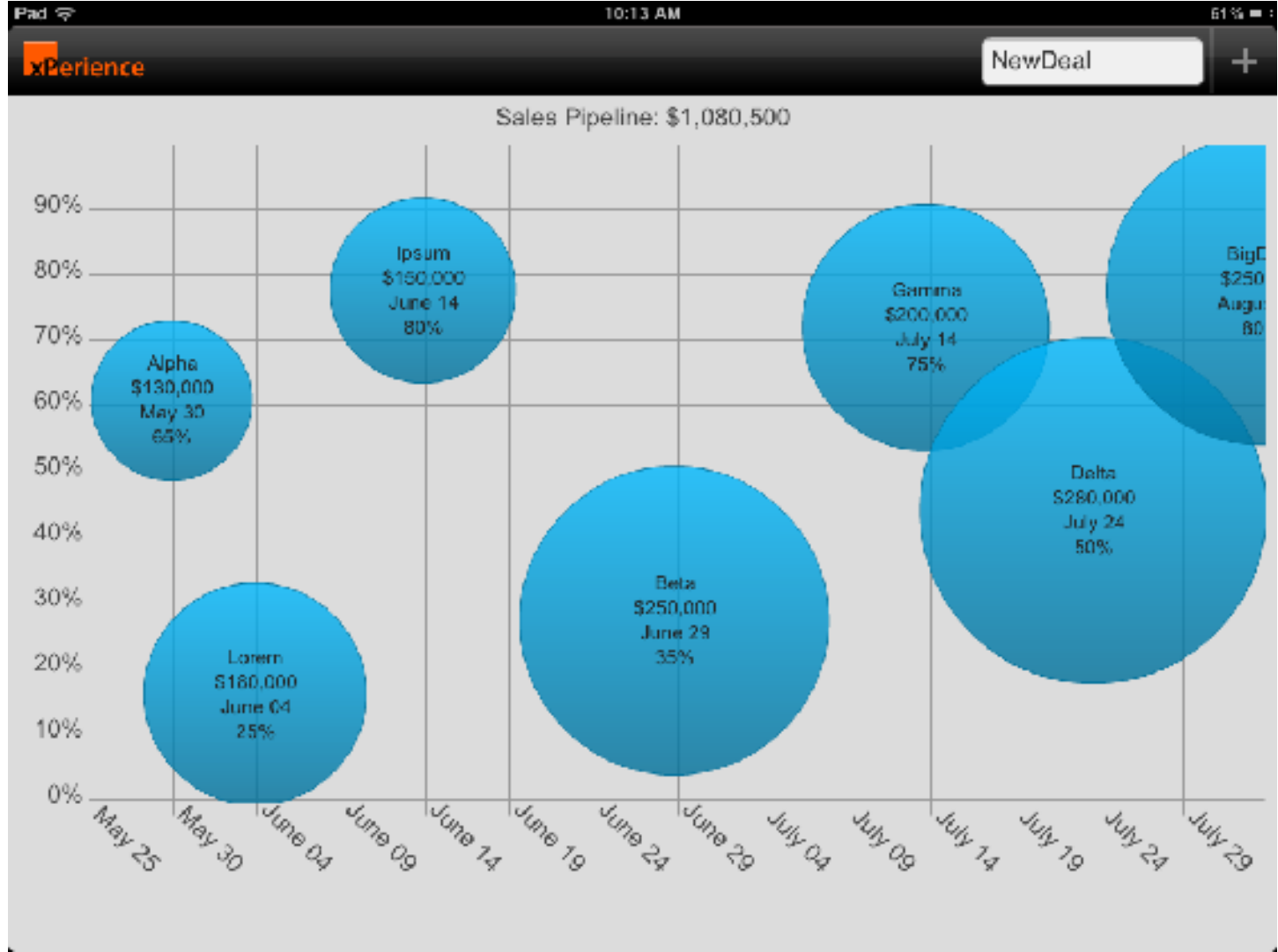
Title centered

Beveled buttons

Per-platform style rules

```
@media (os-platform: "ios") {  
  ActionBar {  
    defaultButtonAppearance: "beveled";  
    titleAlign: "center";  
  }  
}
```

Demo



Demo



