FIRE PHONE DEVELOPMENT

Jia Tse & Matthew Runo

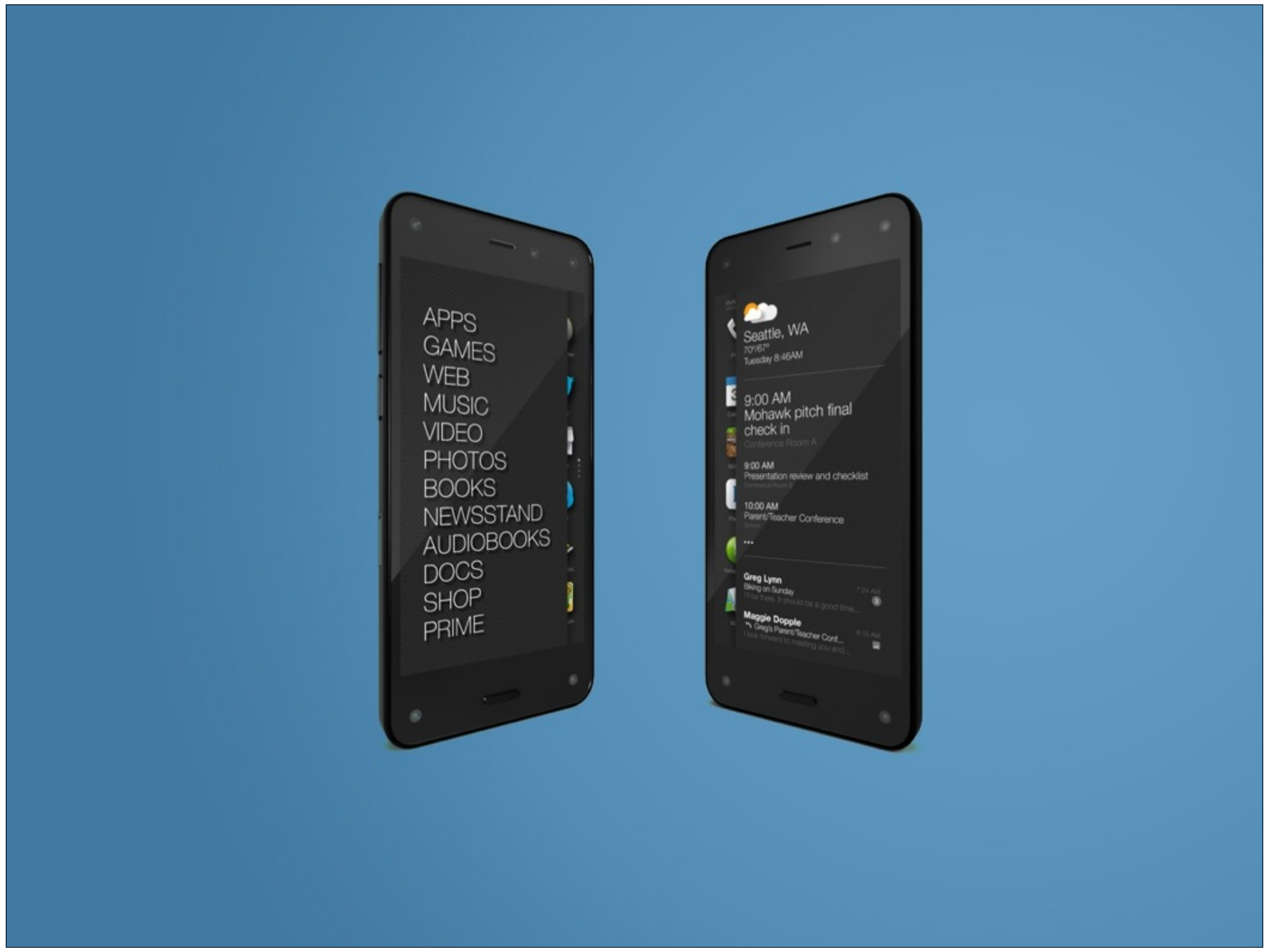
Email
4
Combined Inbox
Tavis Blanc
10:24 AM
Pea Patch Book
I am thinking of picking one of these up to help with the Pea Patch. Any...
Frank Ruault
9:58 AM
All Hands Meeting
Don't forget: we have an All-Hands on Monday morning with marketing and...
Dialer
Messaging
Email
Silk

FIRE PHONE SDK

—Amazon Fire Phone SDK, freely available.

—Headtracking, Motion Gestures, 3D controls

—Android 4.2.2, KitKat coming.



—Left Panel is basically an expandable list, but custom for the Fire Phone

—Contains application navigation and controls. Selecting something here should change the content of the main panel

—You are restricted to the default look, unless you want to do lots of custom code.

On the other hand, the Right Panel can be anything you want it to be.

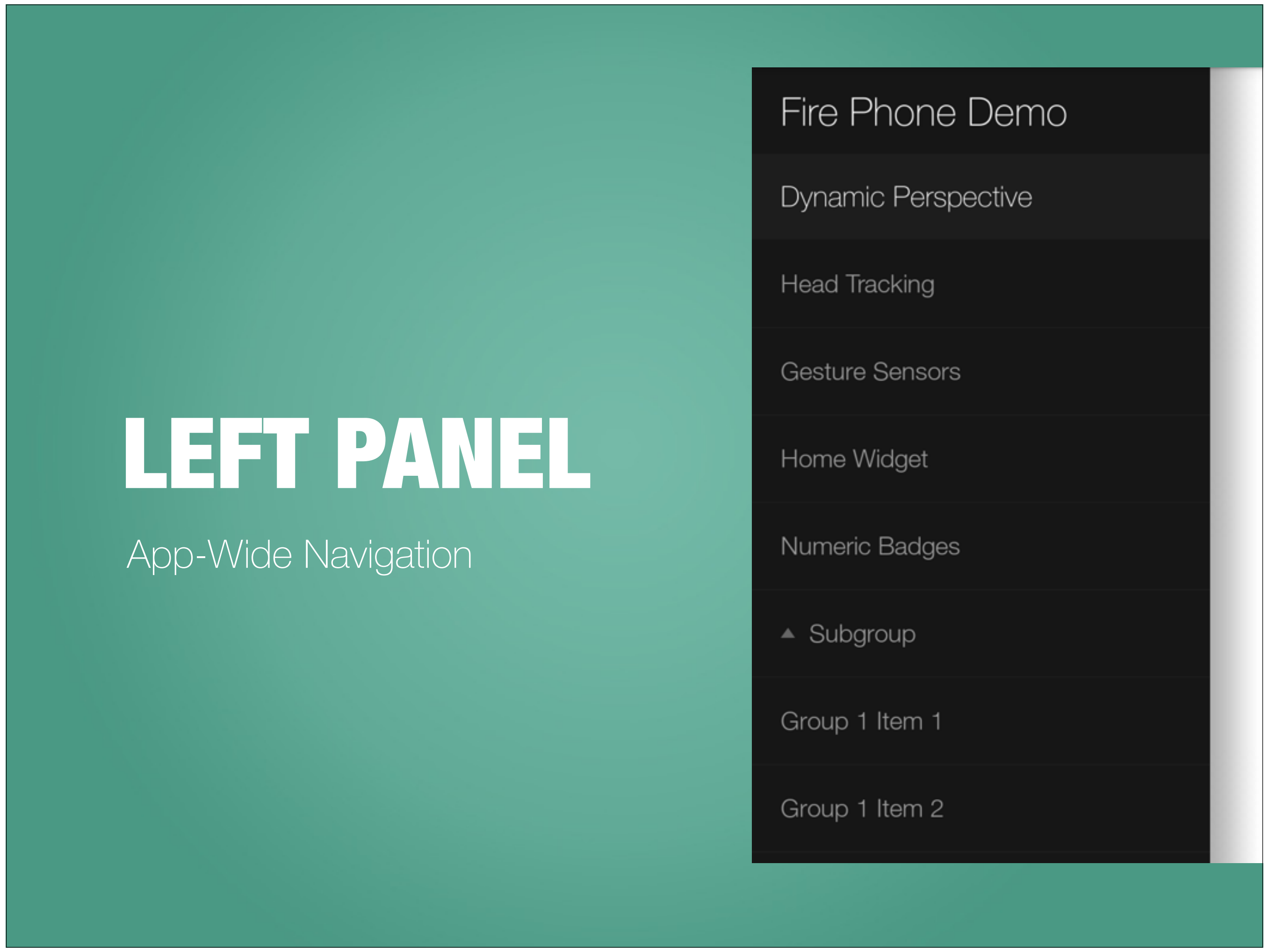
— Right panel is for content that will enrich the experience for your users. It encourages serendipity: “I didn’t know I needed that, but it’s perfect!”

```
<amazon.widget.SidePanelLayout
  amazon:leftPanel="@+id/left_panel"
  amazon:content="@+id/main_panel"
  amazon:rightPanel="@+id/right_panel" >

  <amazon.widget.NavigationPane
    android:id="@id/left_panel"
    amazon:navMenu="@menu
      menu_sidepanel_navigationpane" />

  <FrameLayout android:id="@id/main_panel" />
  <LinearLayout android:id="@id/right_panel" />

</amazon.widget.SidePanelLayout>
```



LEFT PANEL

App-Wide Navigation

Fire Phone Demo

Dynamic Perspective

Head Tracking

Gesture Sensors

Home Widget

Numeric Badges

▲ Subgroup

Group 1 Item 1

Group 1 Item 2

—This is an example of the left panel.

—(Dynamic Perspective is a nav header)

—Everything else is called nav navitem

—This is a nav group. When clicked, it will expand to review more nav items within that group

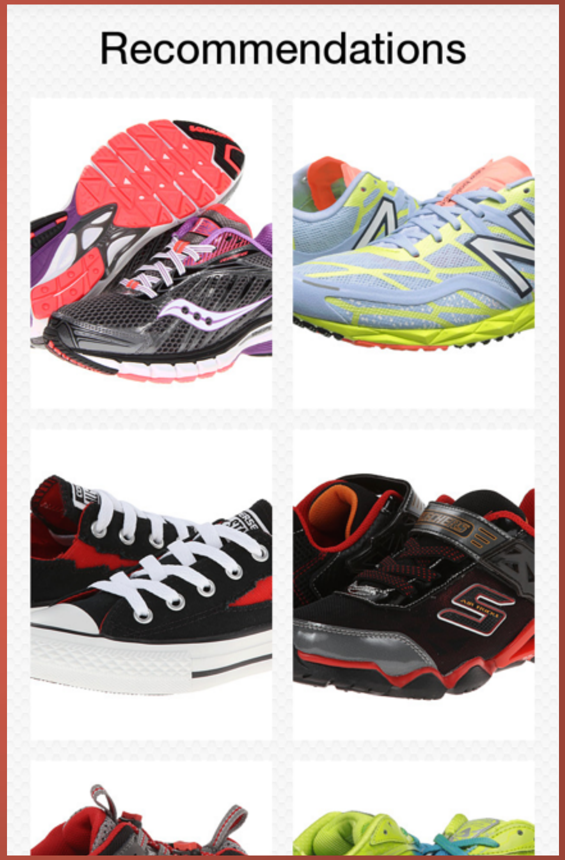
```
<navmenu>
  <navheader android:title="Dynamic Perspective”/>

  <navitem android:title="Head Tracking"/>
  <navitem android:title="Gesture Sensors”/>

  <navgroup android:title="@string/nav_subheader">
    <navitem android:title=“@string/group1_item1” />
    <navitem android:title=“@string/group1_item2" />
  </navgroup>
</navmenu>
```

RIGHT PANEL

Contextual Information



—Contextual information

—Activated by tilting the phone, or flicking your head

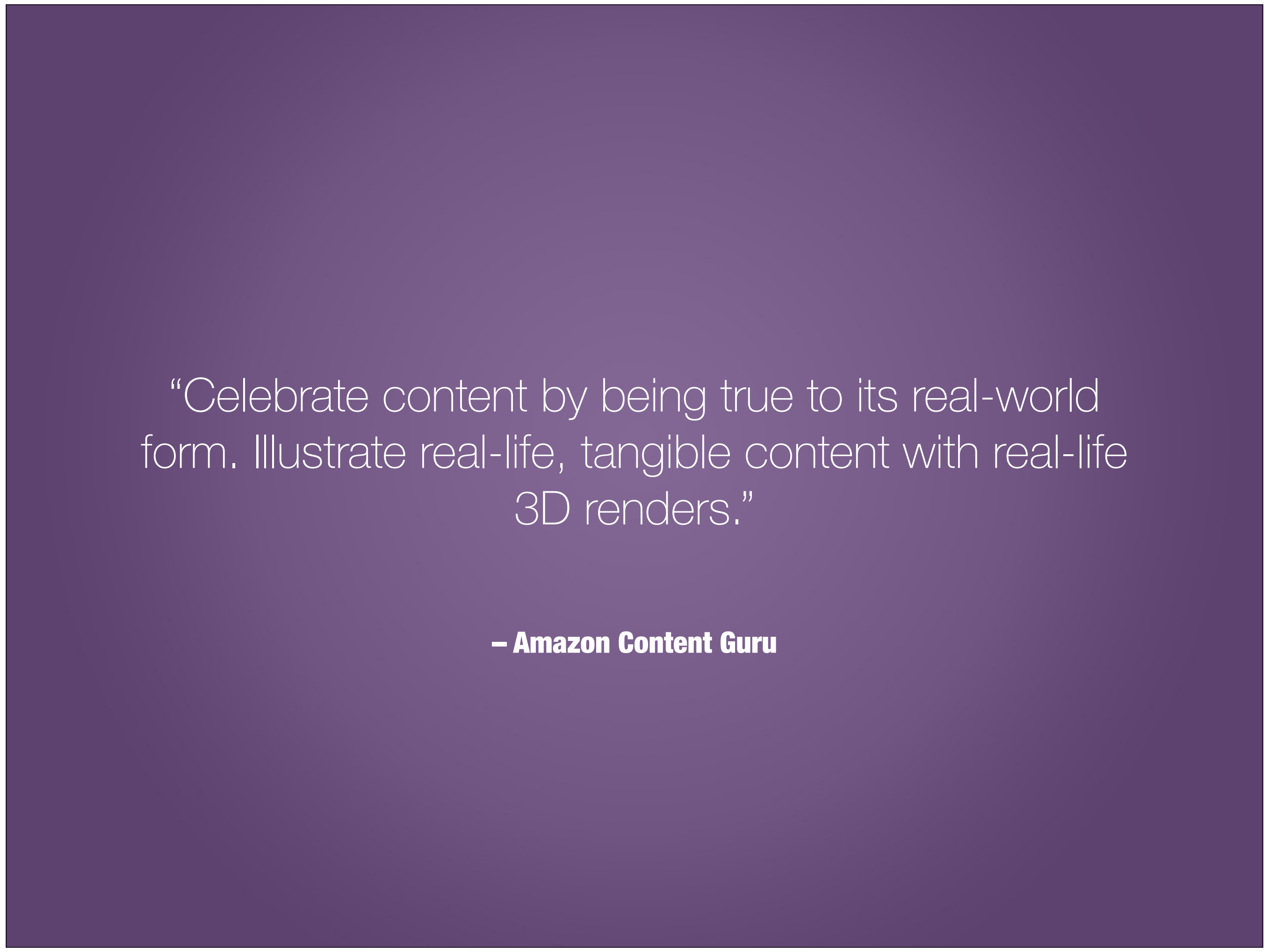
…transition to next slide

DESIGN PRINCIPALS

MATT

When thinking about apps, this is what Amazon would say you should focus on.

These design principal quotes are loosely credited to Jeff Bezos.



"Celebrate content by being true to its real-world form. Illustrate real-life, tangible content with real-life 3D renders."

– Amazon Content Guru

Of course, this assumes you know how to use Maya to create 3d models. If you don't have a game development background, this will be hard to do.

DYNAMIC PERSPECTIVE

—… and that is called dynamic perspective.

—The technology that allows the Fire Phone to respond to the way a user holds, views and moves the devices.

—This is what makes the Fire Phone different than any other android device. Work off of the head tracking cameras plus other sensor data.

ICONS

Can be 3d objects, but you have to deal with creating them from scratch

Should be created in two sizes - large and detailed and smaller and less detailed.

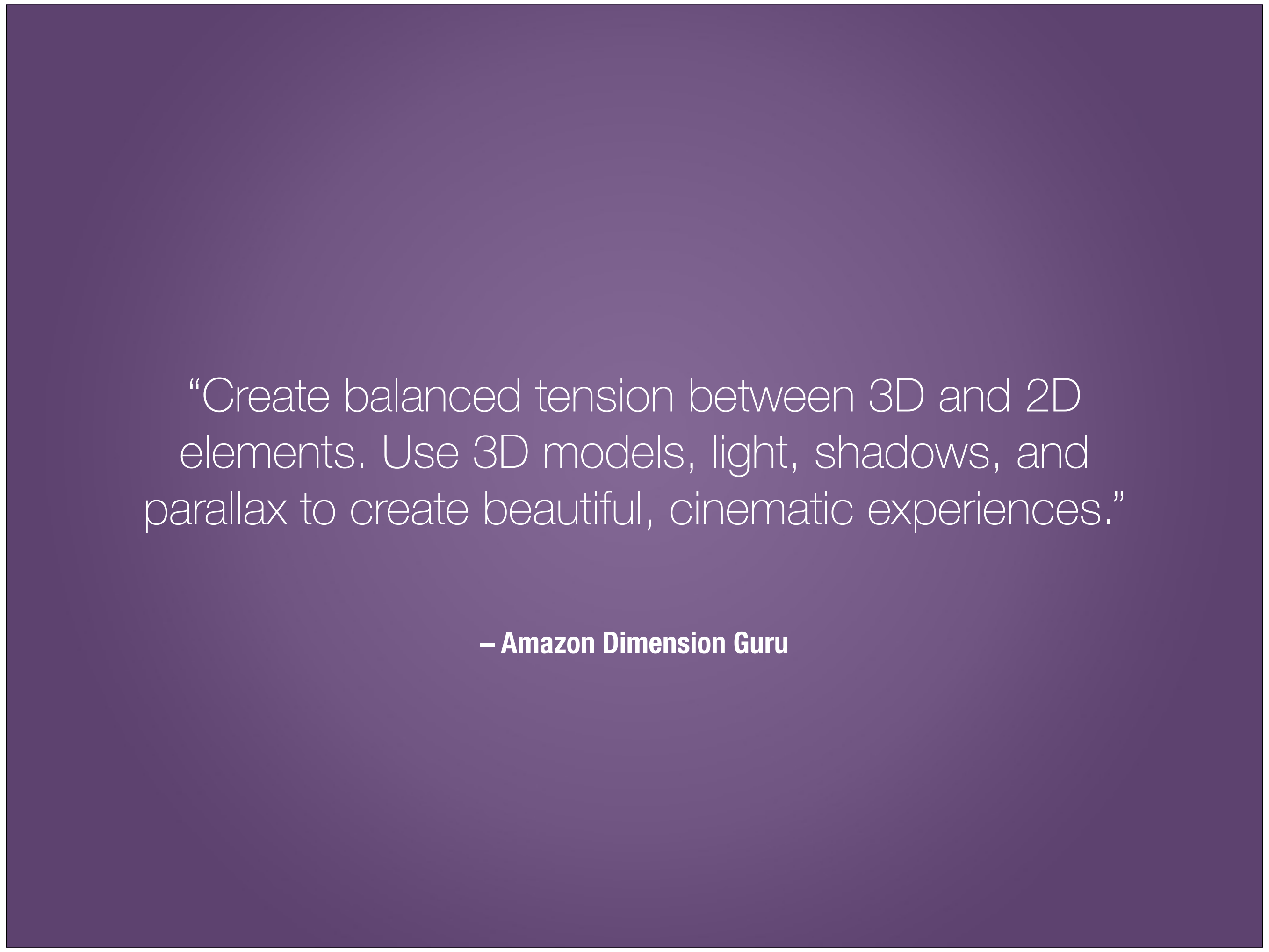
No 3d icon? The phone will use a standard android icon instead.

AUTODESK MAYA

3DSMax is also an option.

Price?

How to learn?


“Create balanced tension between 3D and 2D elements. Use 3D models, light, shadows, and parallax to create beautiful, cinematic experiences.”

– Amazon Dimension Guru

MATT

Your apps do not have to be flat and one dimensional. The phone can tell where the user is looking, and your UI can react to that and engage the user in much more interesting ways than on other devices.

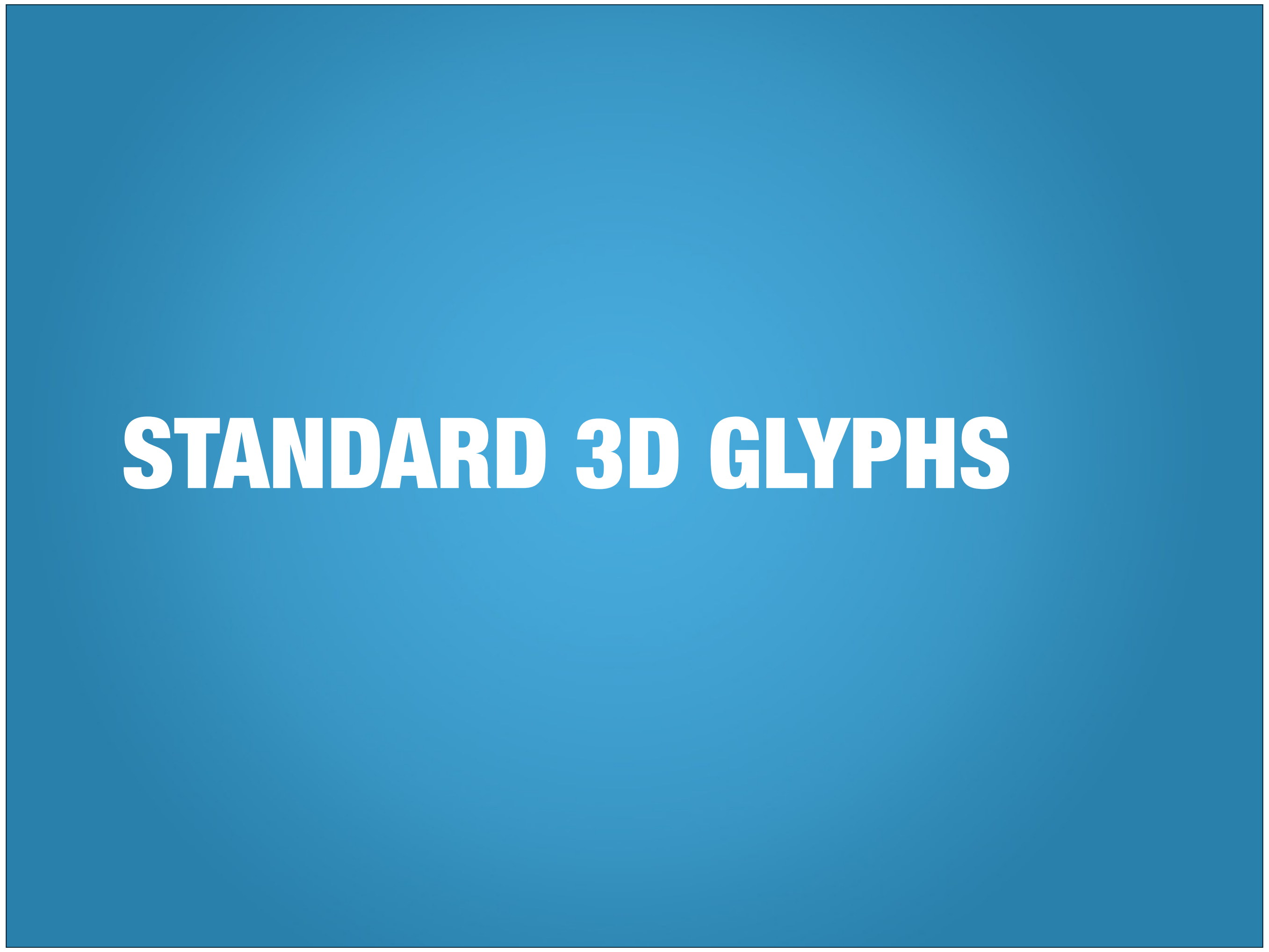
ACTION BARS

In Fire Phone - Header Navigation Bars.

Includes up to 2 action controls, plus a title and optional up navigation button.

Comes in 3d or 2d flavors, depending on what you want to do.

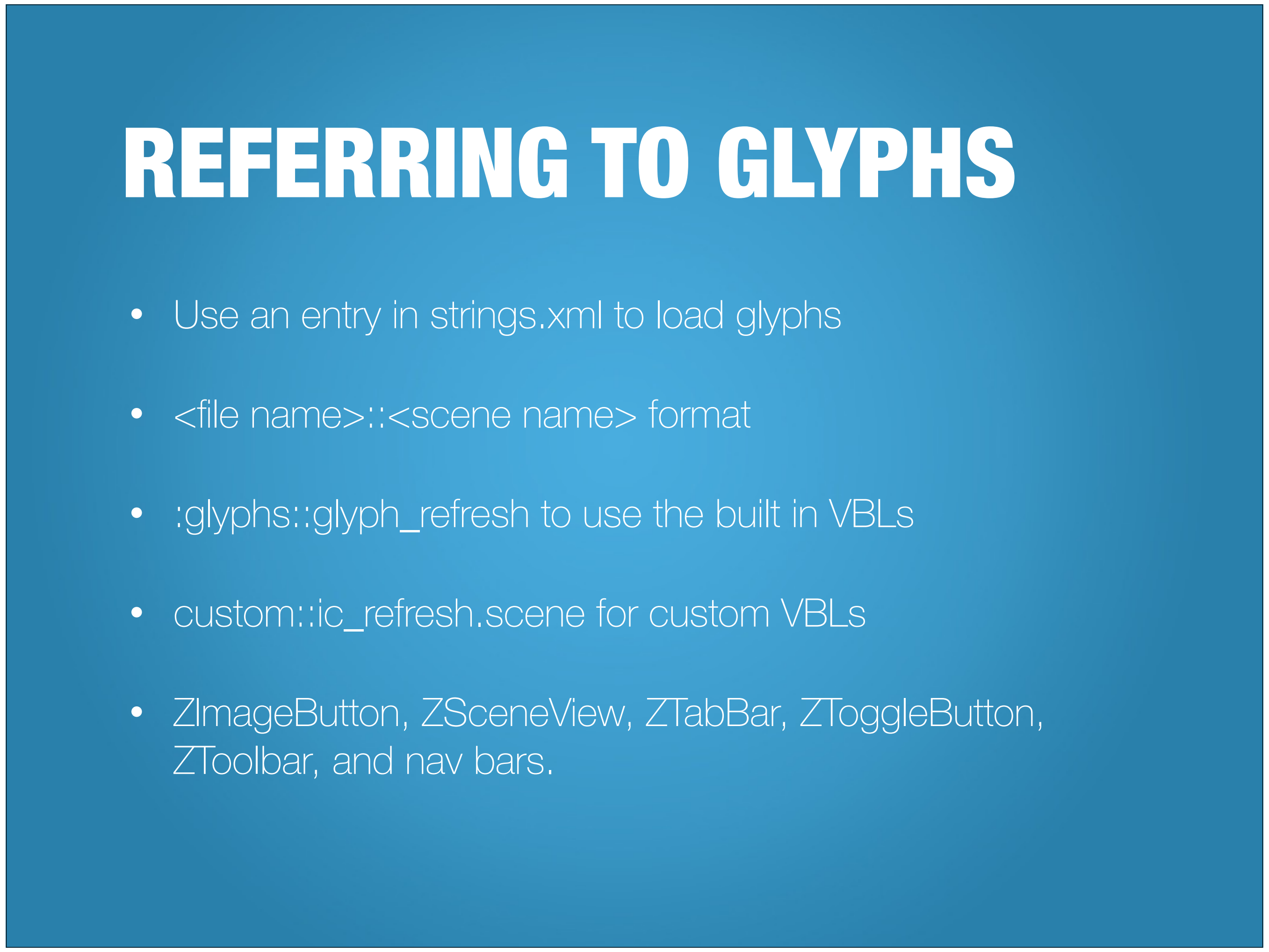
You cannot put a 2d icon inside a 3d header navigation bar.



STANDARD 3D GLYPHS

There are lots of standard 3d glyphs. Most of the standard android ic_action icons are available as a 3d glyph, so you might not need to do any 3d modeling!

https://developer.amazon.com/public/solutions/devices/fire-phone/docs/incorporating-glyphs-into-your-app



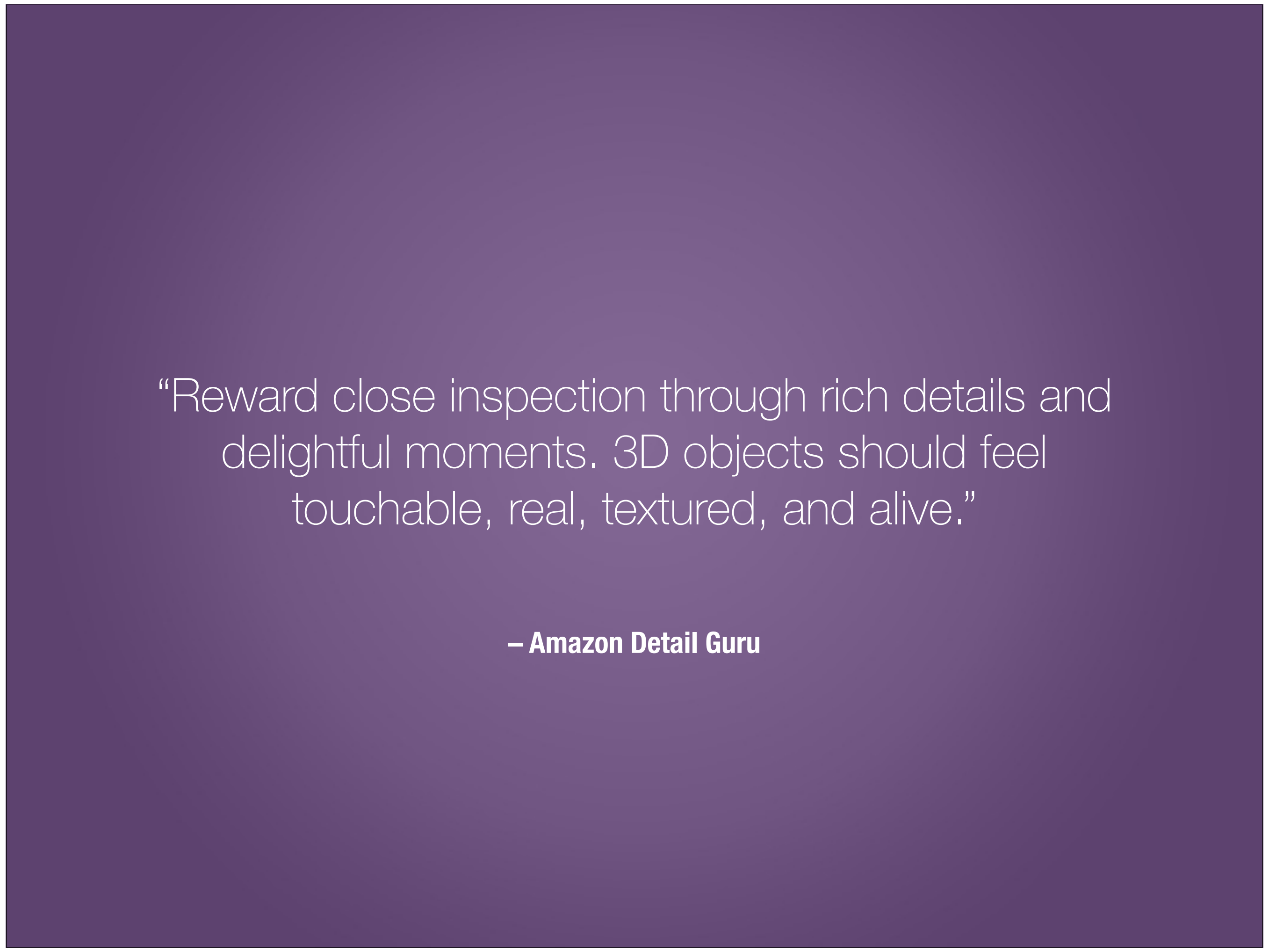
REFERRING TO GLYPHS

- Use an entry in strings.xml to load glyphs
- <file name>::<scene name> format
- :glyphs::glyph_refresh to use the built in VBLs
- custom::ic_refresh.scene for custom VBLs
- ZImageButton, ZSceneView, ZTabBar, ZToggleButton, ZToolbar, and nav bars.

You need to refer to the glyph you want to use in strings.xml using a specific format, and then use the string entry in your XML android:icon="" entry.

Directly set them using euclid:vbl and euclid:scene

CODE SAMPLE

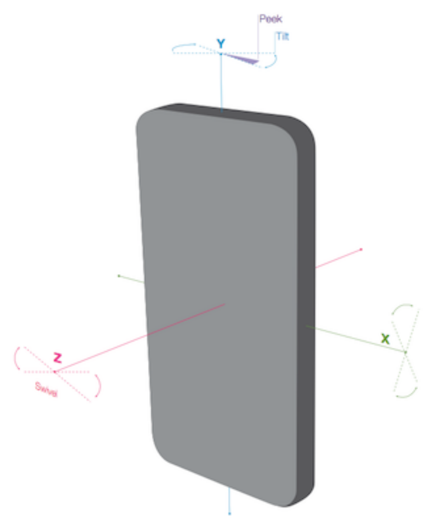

"Reward close inspection through rich details and delightful moments. 3D objects should feel touchable, real, textured, and alive."

– Amazon Detail Guru

JIA

The longer a user spends in your app, the more they can discover. Let them look around objects, put surprises where people can find them to reward their interaction.

MOTION GESTURES



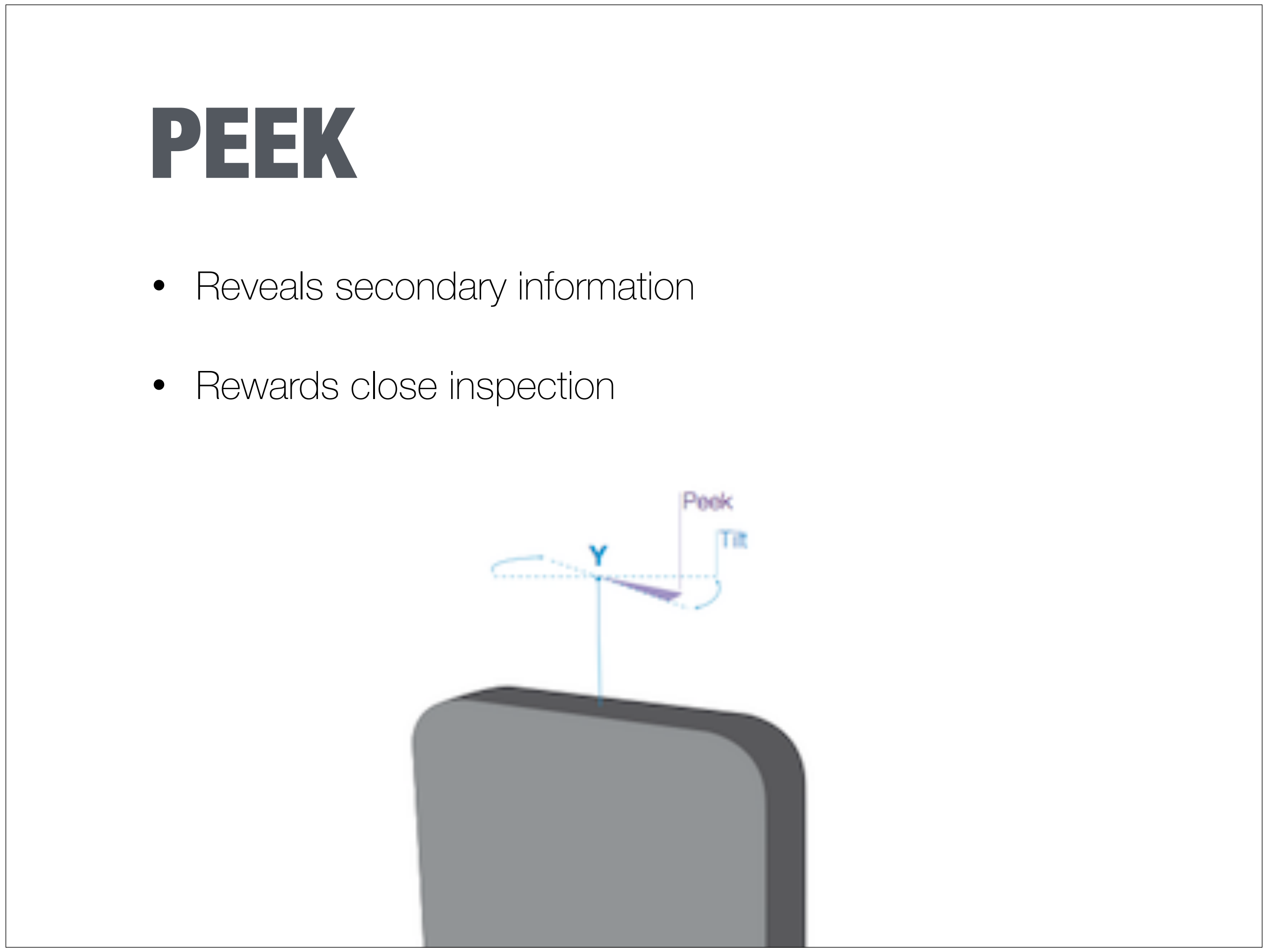
PEEK, TILT, AND SWIVEL

Motion Gestures

Peek - A small rotation around the Y axis.

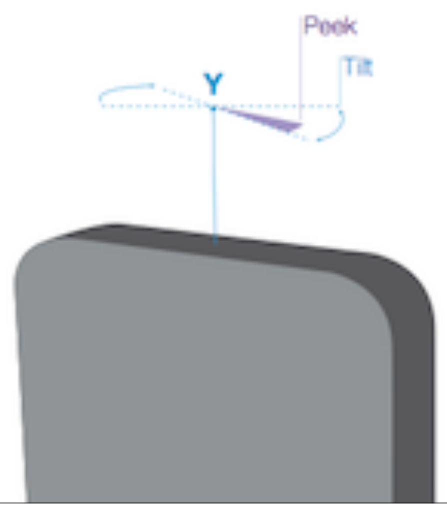
Tilt - A big rotation around the Y axis.

Swivel - A rotation around the Z axis.



PEEK

- Reveals secondary information
- Rewards close inspection



—Peek is a small rotation on the y-axis.

—When your head moves slightly off-center in relation to the device, a peek is triggered.

—Peek is designed to show you things in the moment. It reveals secondary information. It rewards close inspection

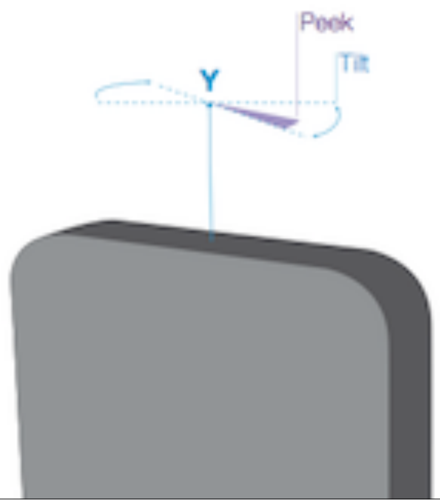
—It is meant to be used on the main panel

—talk more about the technical

  —tells you when it’s peeking, and leaving peek, and direction it came from

TILT

- Primarily used to access left and right panels
- Can override for custom functionality

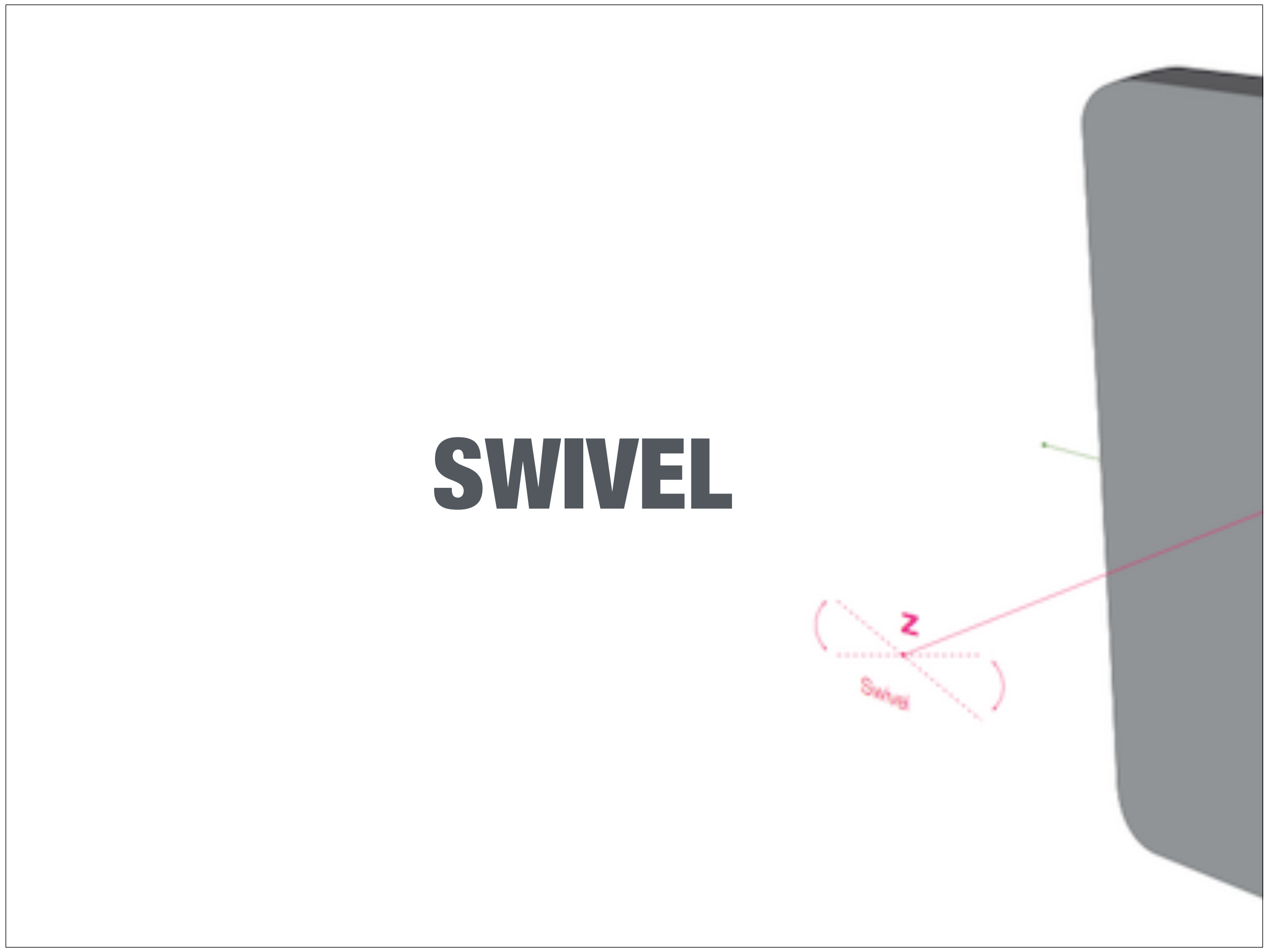


A big rotation around the Y axis. Tilt is used to show and hide the side panels.

—Tilts are basically gesture events, so you can override them and perform custom actions if you want.

—talk more about the technical

—tilt, and direction. Forward to scroll if you want.



SWIVEL

Not directly exposed for use. Used by the system to show system UI, notification drawer, etc.

SETUP

- `GestureManager.createInstance(Activity)`
- `manager.registerListener(Activity, peekGesture, GestureEvent.Direction.LEFT | GestureEvent.Direction.RIGHT)`
- `onGestureEvent(GestureEvent)`

—Basically there are 3 steps to get head tracking setup for your app

—Create an instance of HeadTrackingManager

—Register a listener or a poller

—then do something when that event happens

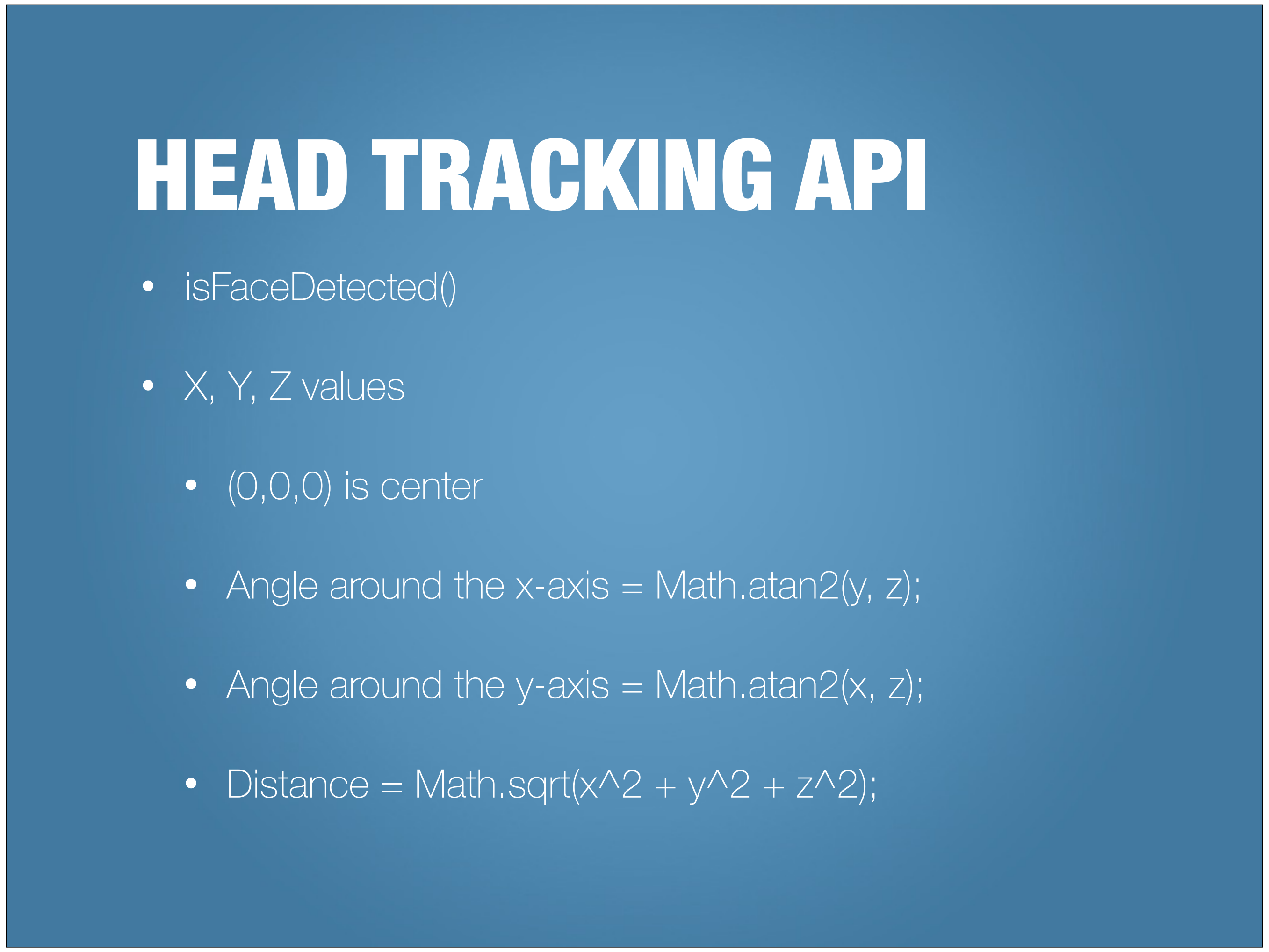
HEAD TRACKING

You get head tracking as an API on the device, but it is also available as a Unity plugin.

Powered by the 4 IR cameras as well as other sensor data.

HEAD TRACKING

- 3D Gaming
- Be Creative



HEAD TRACKING API

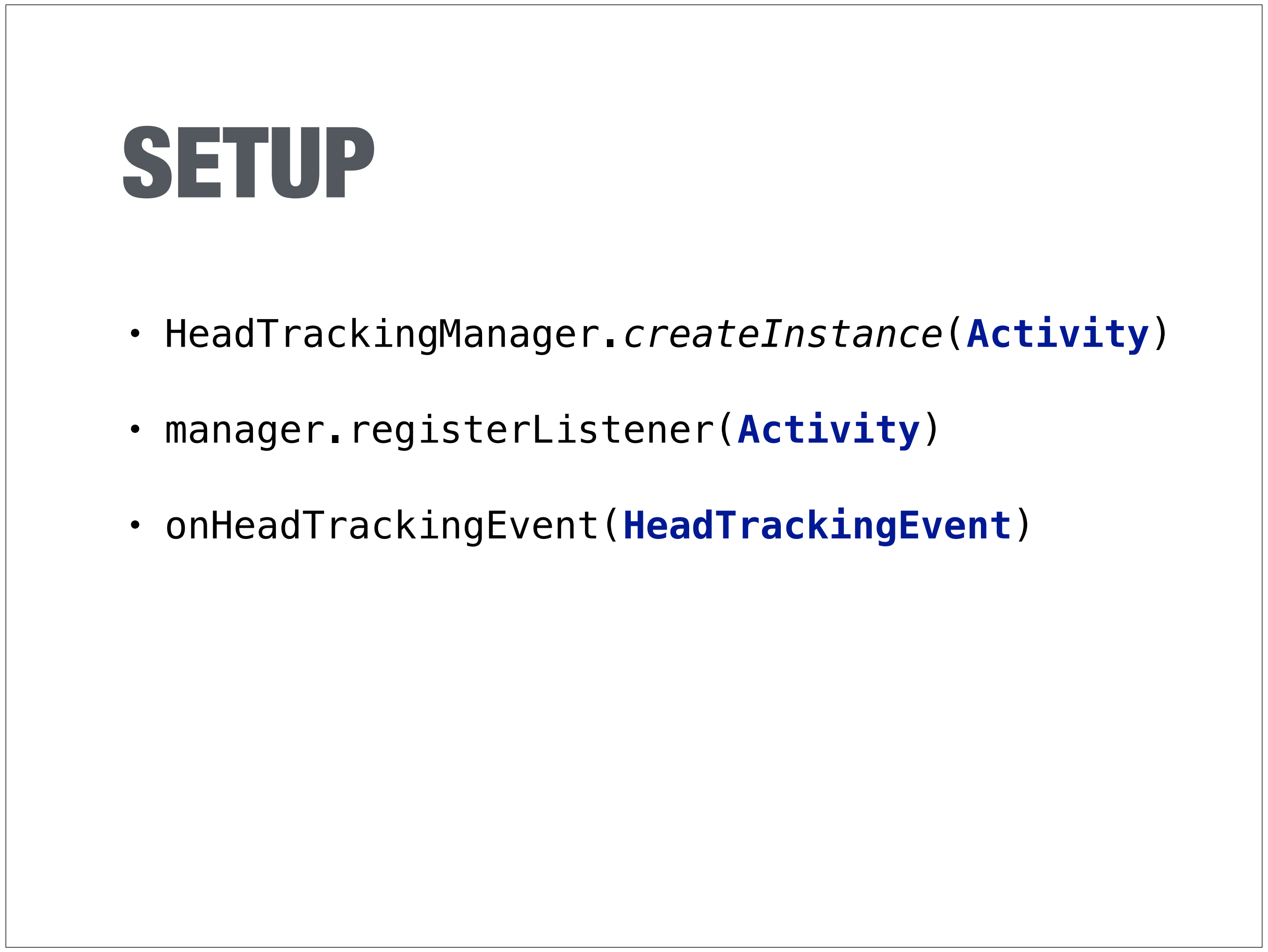
- isFaceDetected()
- X, Y, Z values
    - (0,0,0) is center
    - Angle around the x-axis = Math.atan2(y, z);
    - Angle around the y-axis = Math.atan2(x, z);
    - Distance = Math.sqrt(x^2 + y^2 + z^2);

—talk more about what the angles are.

—(0,0,0)

HEADTRACKINGPOLLER
&
HEADTRACKINGLISTENER

There are 2 ways to get head tracking events. Polling for data or listening for data.



SETUP

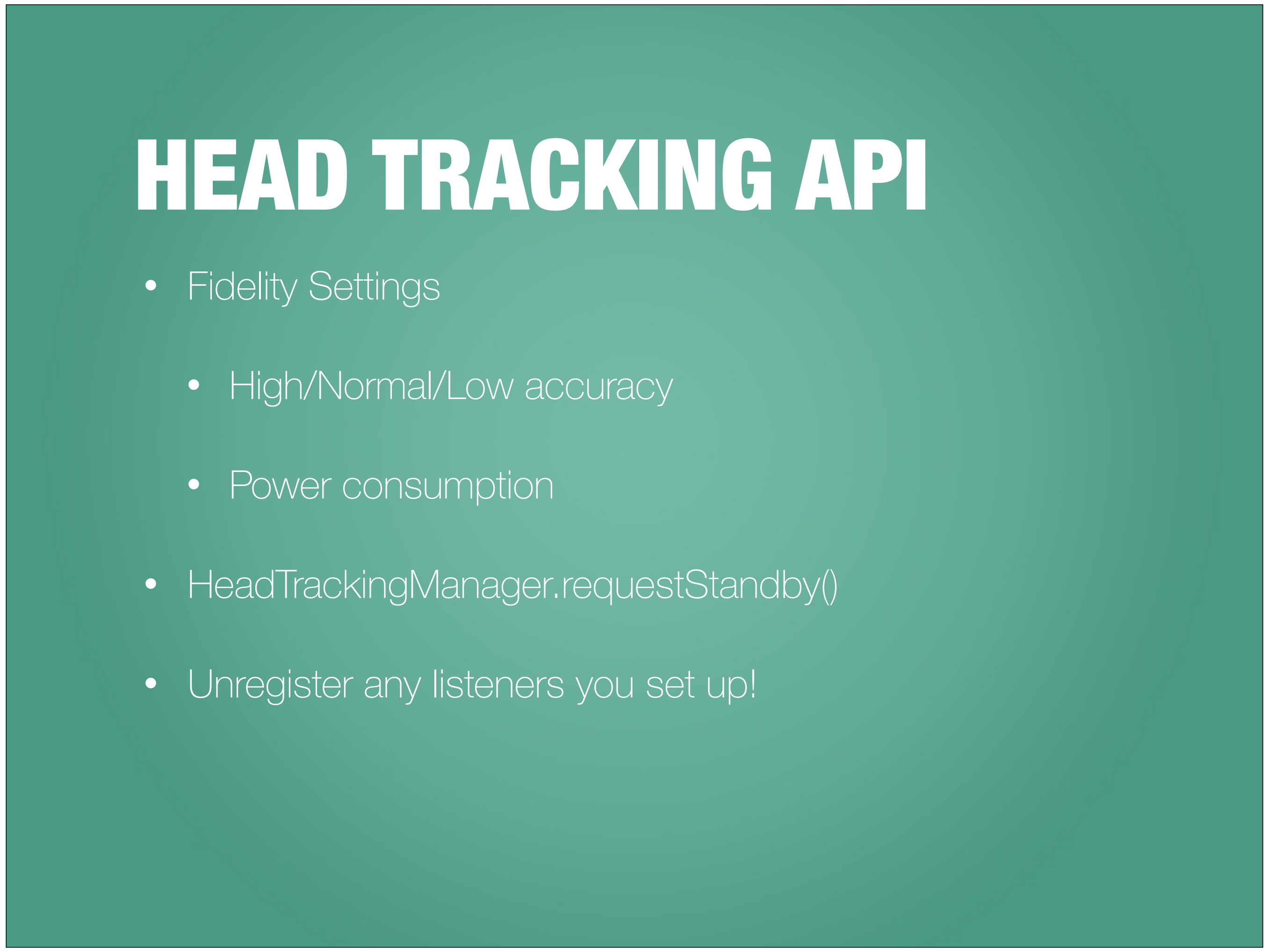
- `HeadTrackingManager.`*`createInstance`*`(`**`Activity`**`)`
- `manager.registerListener(`**`Activity`**`)`
- `onHeadTrackingEvent(`**`HeadTrackingEvent`**`)`

—Basically there are 3 steps to get head tracking setup for your app

—Create an instance of HeadTrackingManager

—Register a listener or a poller

—then do something when that event happens



HEAD TRACKING API

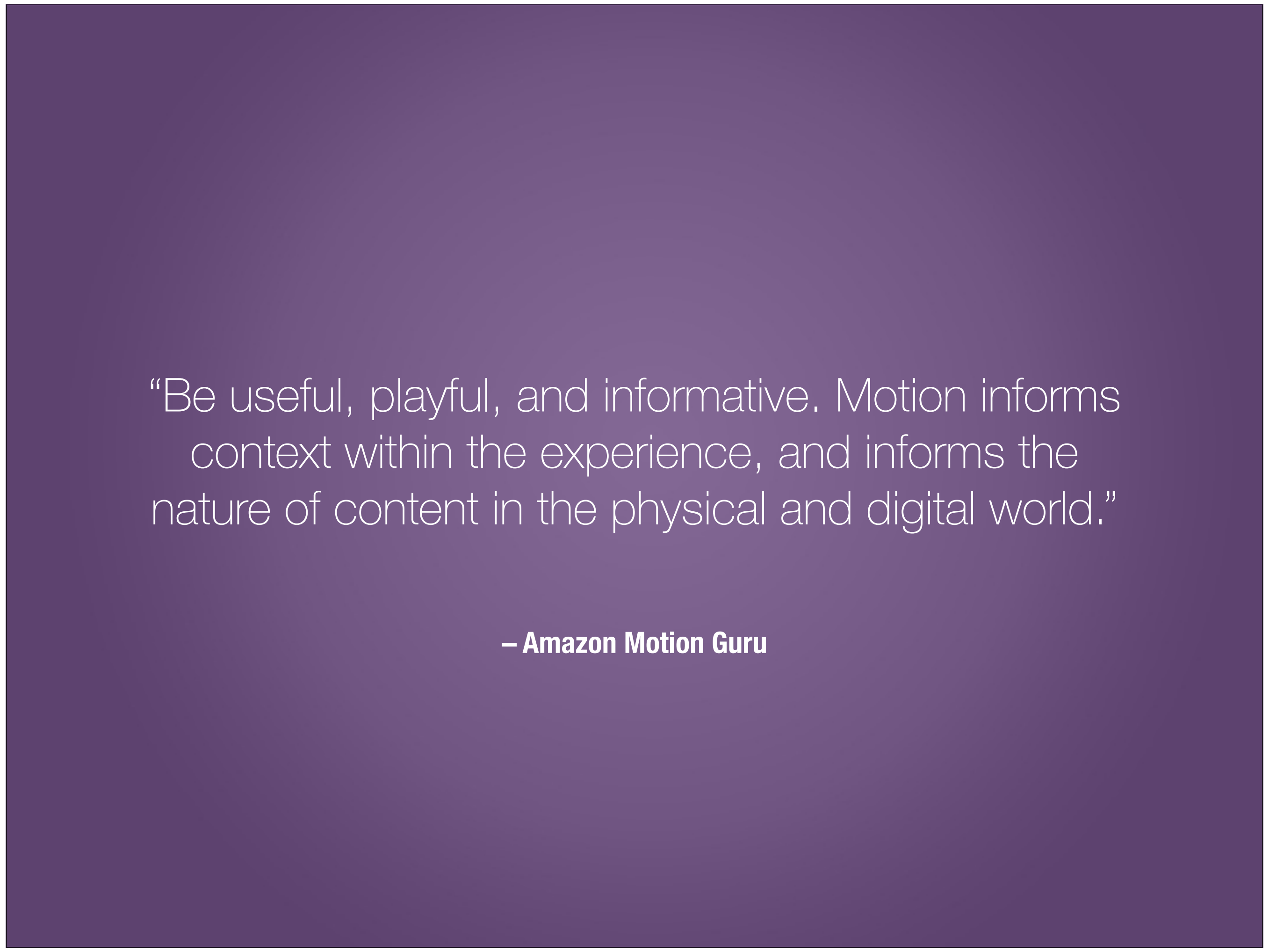
- Fidelity Settings
  - High/Normal/Low accuracy
  - Power consumption
- HeadTrackingManager.requestStandby()
- Unregister any listeners you set up!

There are 3 ways to save energy.

—The first is to mess with the accuracy settings. The higher the accuracy, the higher the power consumption

—If head tracking is not immediately needed, it can be put into standby mode. If it's totally not needed (such as when an activity pauses, be sure to unregister any listeners or pollers)

— (show fidelity levels on head tracking sample)


"Be useful, playful, and informative. Motion informs context within the experience, and informs the nature of content in the physical and digital world."

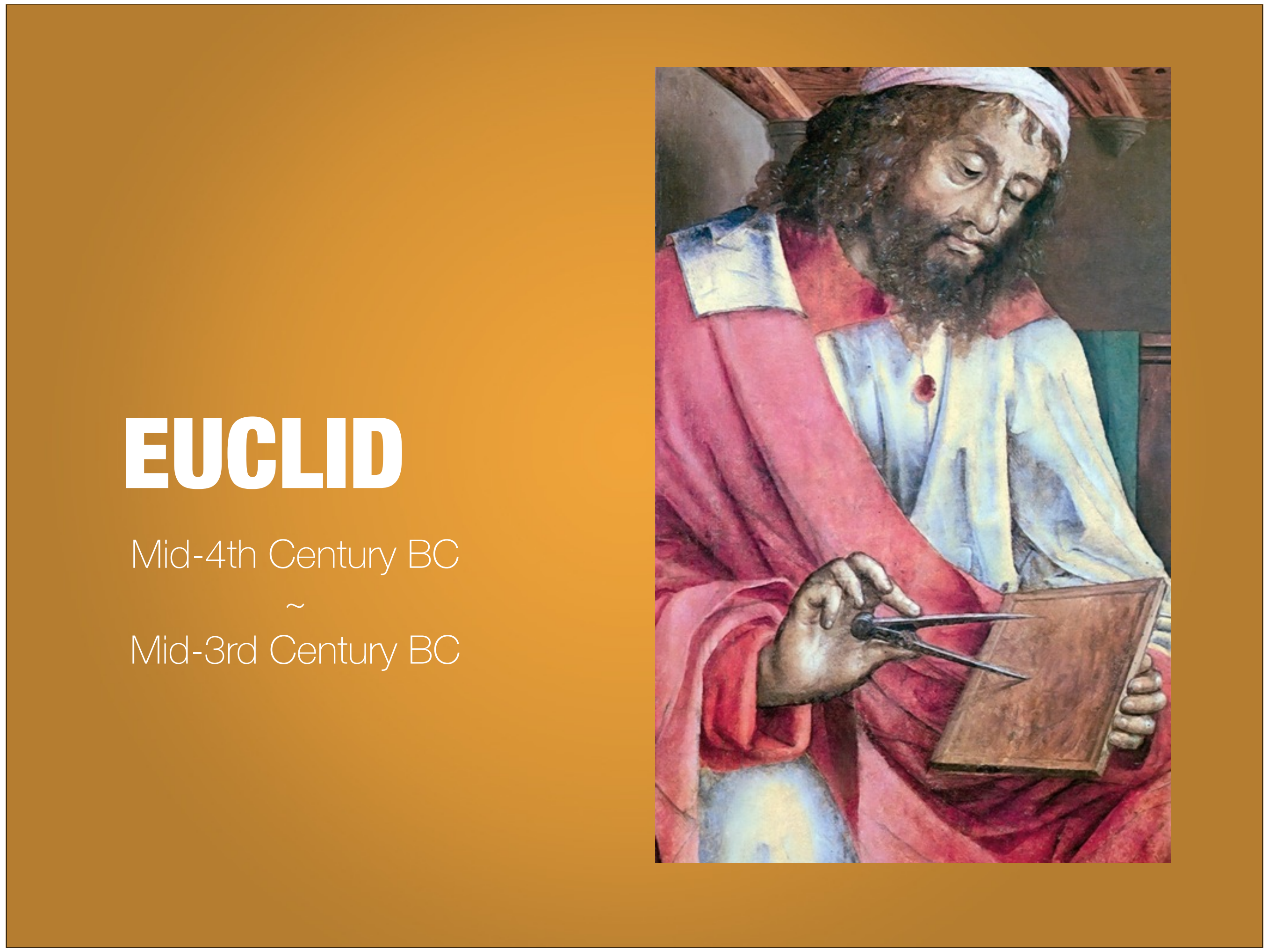
– Amazon Motion Guru

MATT

Use animations throughout your app. Just like in Android L, animations connect screens and make your app experience that much more fluid and memorable.

3D LAYOUTS

Layouts that move



EUCLID

Mid-4th Century BC
~
Mid-3rd Century BC

Namespace to import to gain access to ZView attributes in XML

ZVIEW?

3D controls are all models created by artists with textures and vertices included.

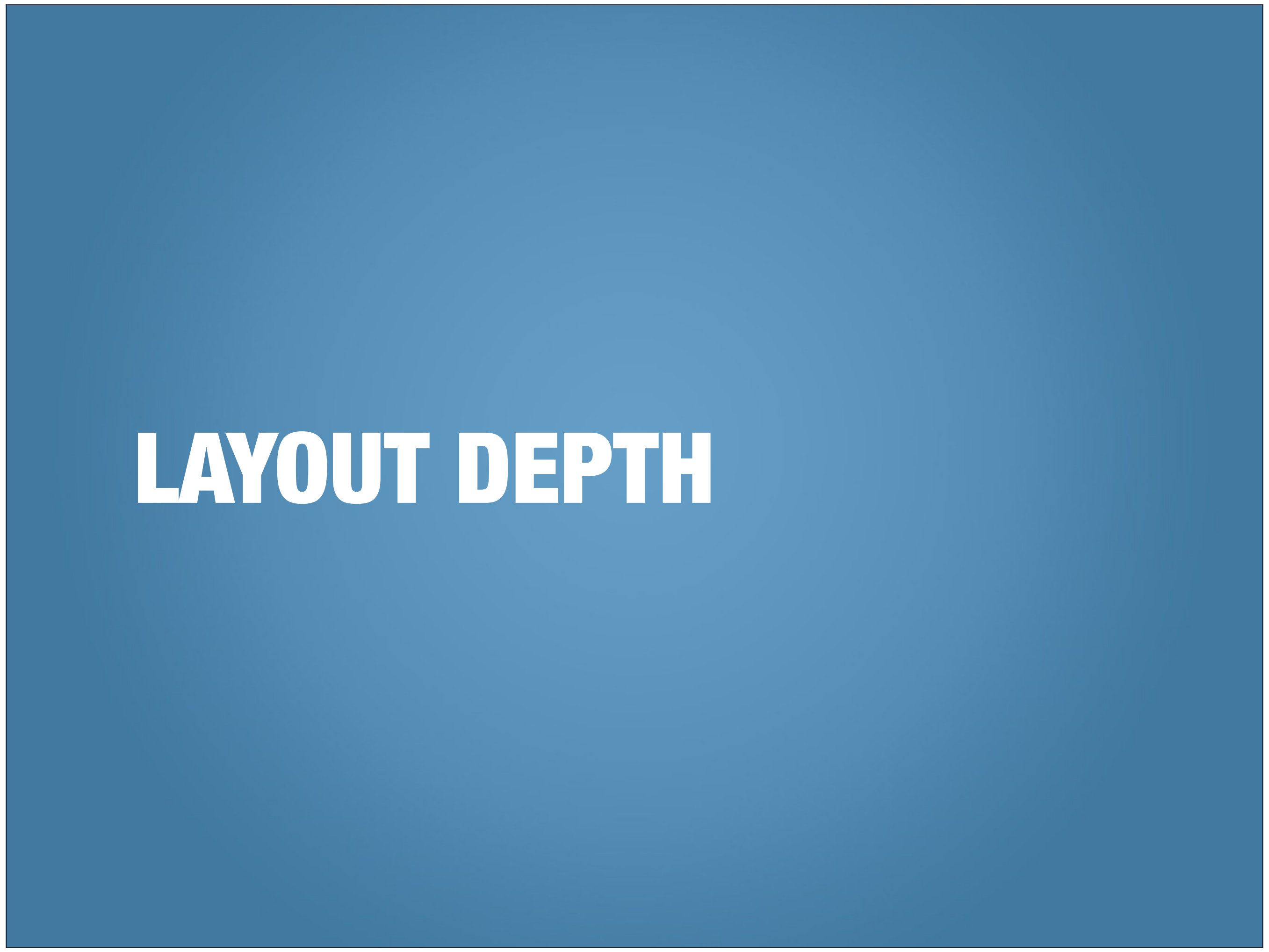
Because of this, all you can really change is the face color of the control.

You cannot add a border, set a custom background, or make changes to it via View.onDraw(). ObjectAnimators DO work though.

ZCONTAINER

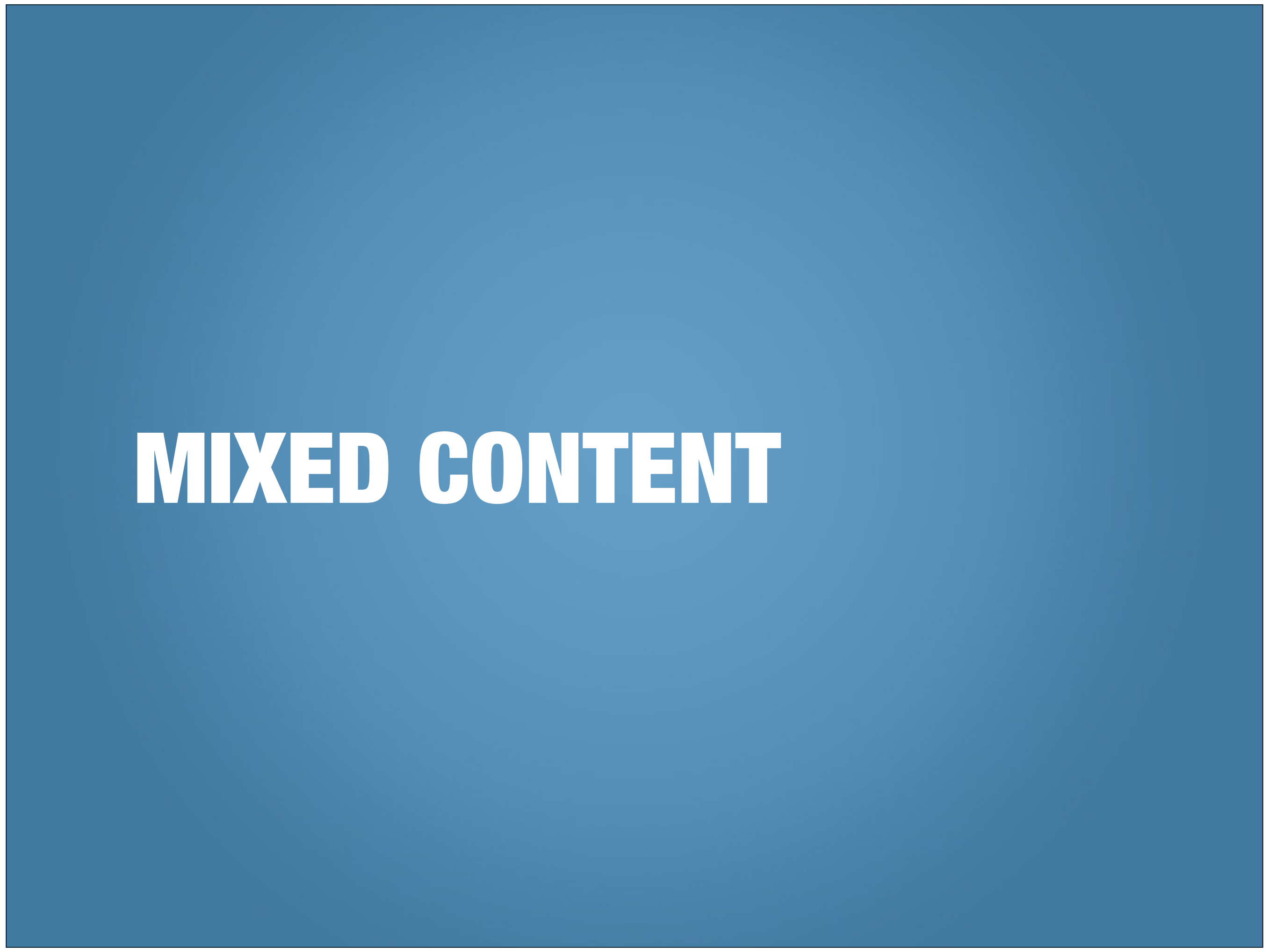
ZContainer is a view that you wrap all your 3d layouts in. It has the effect of enabling the OpenGL rendering engine, and also floats wrapped content by 2mm.

Autopadding.. ?



LAYOUT DEPTH

Euclid’s layout_depth works the same here as in Android L. Use it to position elements above or below other euclid views.



MIXED CONTENT

Generally you cannot mix 2d and 3d layout views. You will get runtime exceptions if you try because the OpenGL renderer can't understand standard android views (remember the special onDraw methods..)

In a couple places you can: inside ZShadowReciever and ZParallaxBackground. Both of these are designed to be roots of your layout.

2D TO 3D?

Linear Layout	ZLinearLayout
Frame Layout	ZFrameLayout
ImageView	ZImageView
Button	ZButton
ListView	ClassNotFoundException

HOME SCREEN WIDGETS

aka Hero Widgets

If you don’t provide a widget, the App Store will fill the space with a “users also bought” style widget.

You don’t really want ads for other apps under your icon, do you?

Email
4
Combined Inbox
Tavis Blanc
10:24 AM
Pea Patch Book
I am thinking of picking one of these up to help with the Pea Patch. Any...
Frank Ruault
9:58 AM
All Hands Meeting
Don't forget: we have an All-Hands on Monday morning with marketing and...
Dialer
Messaging
Email
Silk

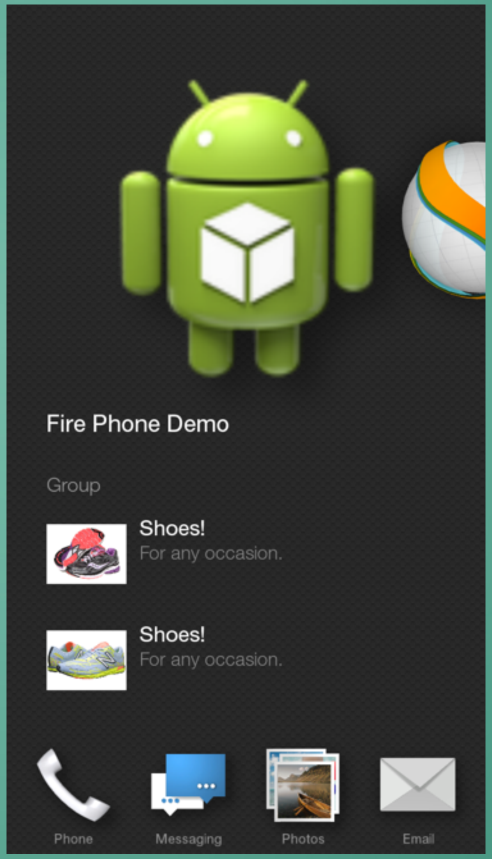
Launcher!

Custom launcher also used on other Kindle devices.

Your app controls two parts of the launcher: your icons, and your home screen widget

LIST WIDGETS

Default, Peekable, Shopping


Fire Phone Demo

Group

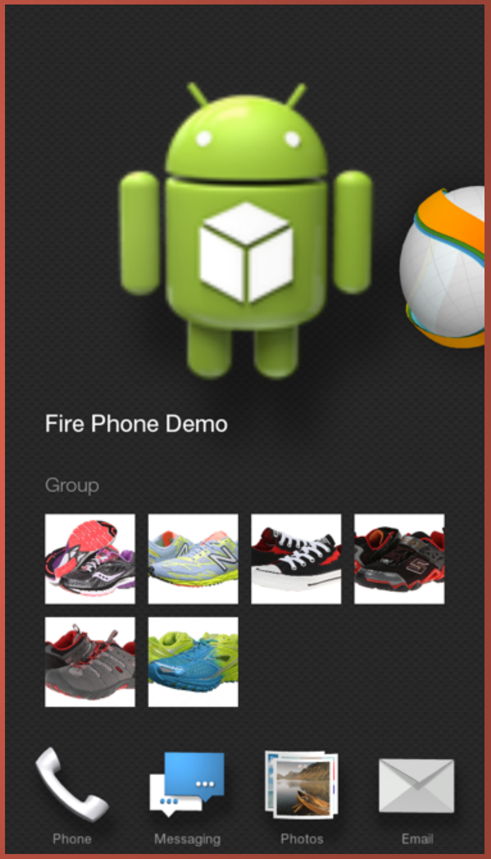
Shoes!
For any occasion.

Shoes!
For any occasion.

Available in a few standard styles: Default, Peekable, Shopping, and Simple

GRID WIDGETS

Only one style!



FIREFLY

Firefly does image and text recognition on the fly. You can create plugins for Firefly which allows it to understand your content as well.

—Recognizes products, books, addresses, phone numbers, music and video.

CODE SAMPLE

FIREFLY

- FireFly Plugin (class)
- DigitalEntityUI (class)
- Final Destination (class)

—Fire fly plugin class - filters

—Digital Entity UI - logic, on click

—Activity

DEMO

GOTCHAS & WORKAROUNDS

MATT

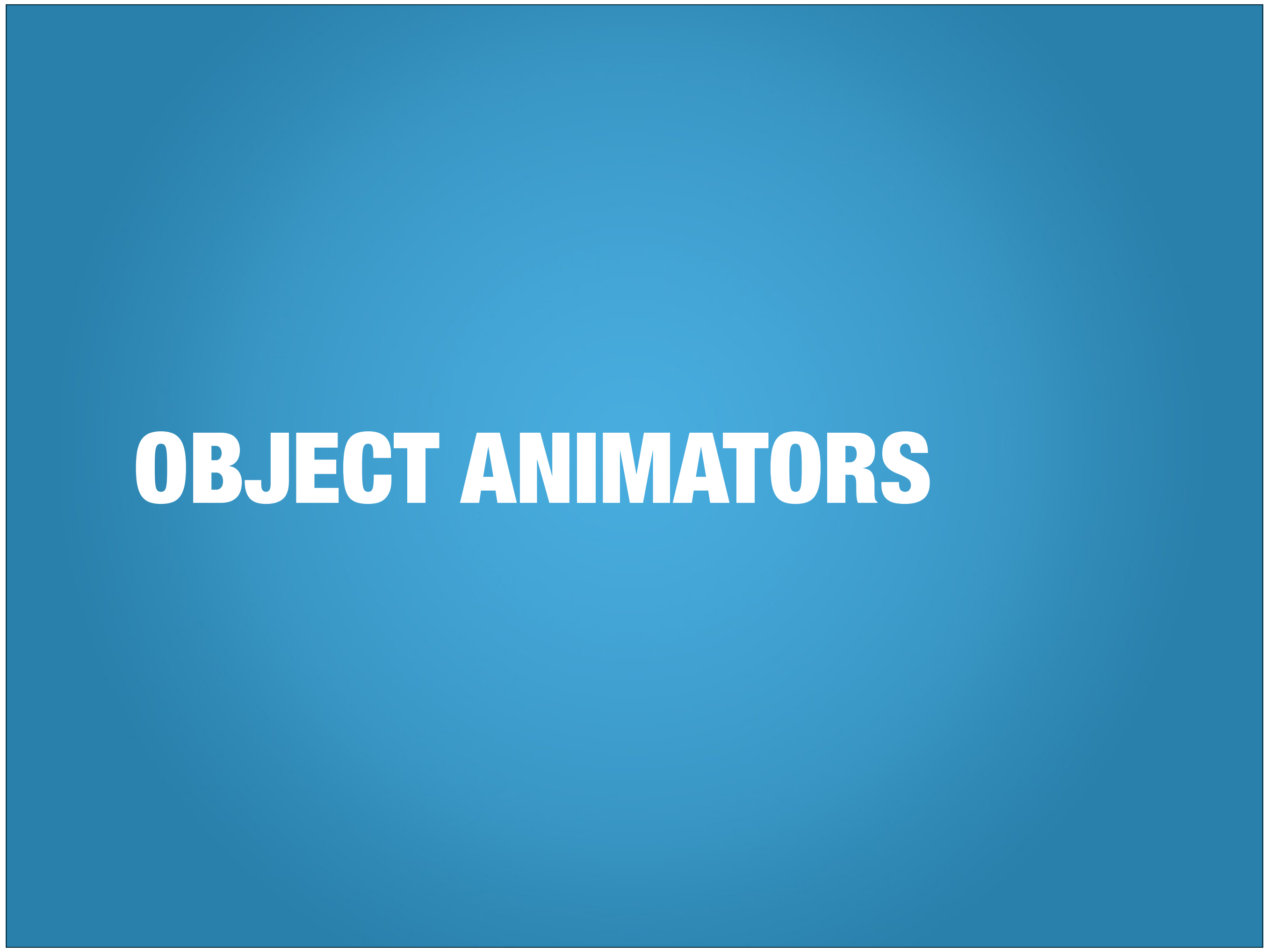
Header Navigation Bars do not support action modes, search views, or custom views. You'll need to fake these in order to have them in your apps - Header Navigation Bar is just a view and can be animated around like any other.

ADAPTERVIEW

Not supported :(

You can to include 3d layouts in your adapters, but that is the extent of it.

Lots of wasted layout layers - nothing you can do.



OBJECT ANIMATORS

These work with 3d views just like you'd expect. You can even link them into custom 3d objects and trigger animations in your objects with in-code animators via attach points.

FRAGMENTS?

Use them, but the main interactions are all activity driven so be aware of that.

ACTION MODES?

Use them - but you'll have to make them custom yourself. This may be coming in a future update to the platform.

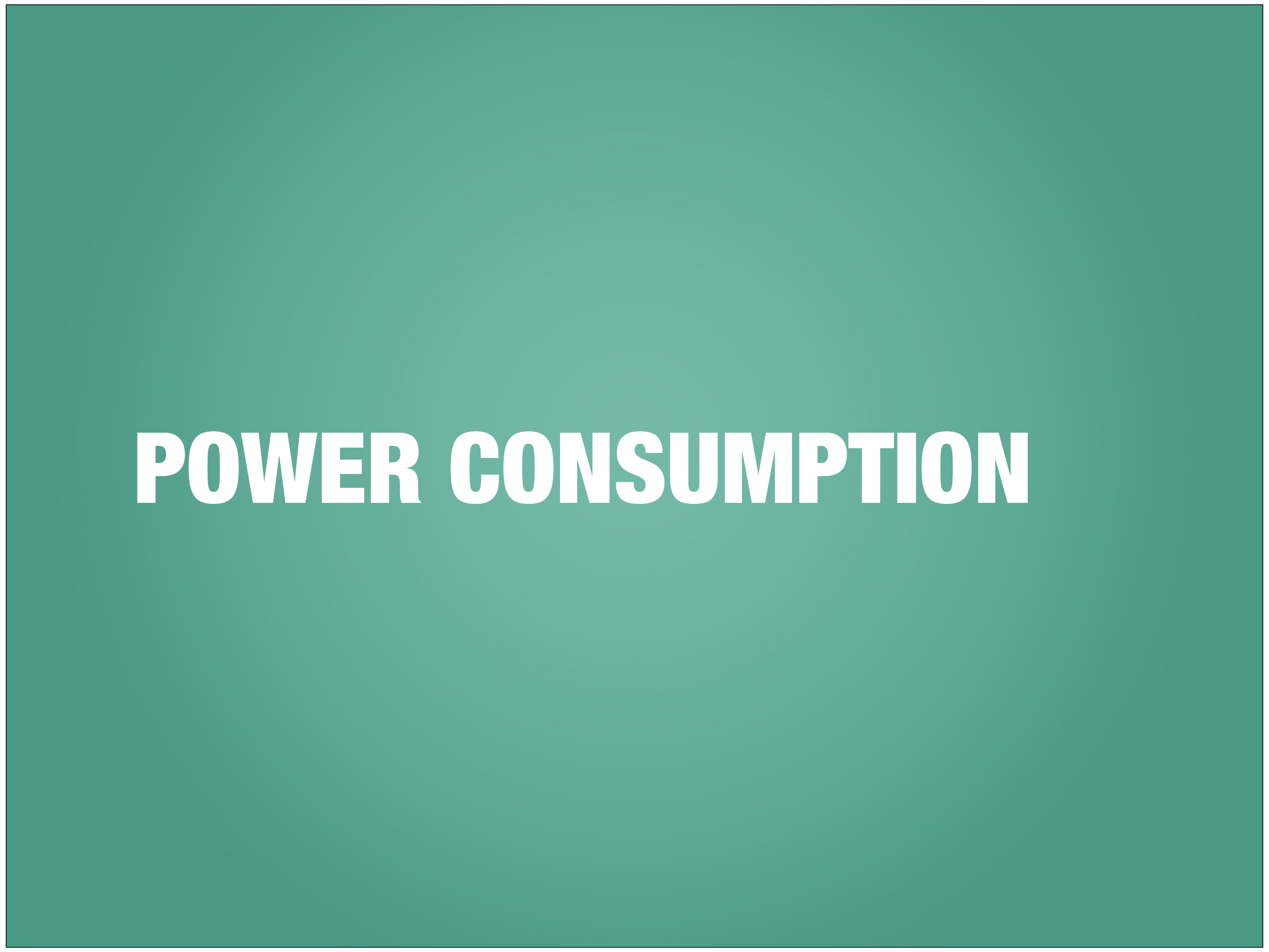
3D ALL THE THINGS?

You can, but good luck… Your UIs will look really strange and be very complex to write.

SHADOWS!

ZShadowReciever is the easiest way to produce shadows, just wrap your 3d views in it. 2d views will not cast shadows!

Text should not cast shadows except in very well thought out cases - it makes it very hard to read.



POWER CONSUMPTION

https://developer.amazon.com/public/solutions/devices/fire-phone/docs/best-practices-for-reducing-power-consumption

HEAT = USED BATTERY

Use the new thermal UI developer option to track how much heat your app generates.

3D APIs and head tracking use a lot of power

THANK YOU

https://github.com/inktomi/FirePhoneDemos