Mobile Application Programming: Android

Custom Controls
Activities

- Apps are composed of activities
- Activities are self-contained tasks made up of one screen-full of information
- Activities start one another and are destroyed commonly
- Apps can use activities belonging to another app
Android Layout

View & ViewGroup

TextView Large Medium Small
Button Small OFF

CheckedTextView
Spinner

Grid
LinearLayout (Vertical)
LinearLayout (Horizontal)
RelativeLayout
FrameLayout
Include Other Layout
Fragment TableRow Space

Button
Text field OFF

Alarm

Sync Browser
5/31/2012 4:58 PM

Address
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor
12:00am
1/1/2011
Custom Controls
3D Color Chooser
Gimbals
The Heist

public class **ImageView**
extends **View**

java.lang.Object
  ➔ android.view.View
    ➔ android.widget.ImageView

- Known Direct Subclasses
  - ImageButton, QuickContactBadge

- Known Indirect Subclasses
  - ZoomButton
Creating a Custom Control

- Create subclass of View class
- Override:
  - `onDraw(Canvas c)`
  - `onMeasure(int wMeasure, int hMeasure)`
- Add listener interface and listener property for the interesting events the control generates and call on... methods when events occur
Creating a Custom Control

Other Considerations:

- Call `setMinimumWidth` and `setMinimumHeight` to give a suggested constraint for the view’s content’s size.
- Consider view state persistence with `onSaveInstanceState`.
Custom Control - **onDraw**

- Draw using provided **Canvas** object
- Information drawn will be **cached** and used by the rendering system to **composite** your control with others on screen in a **performant** way
- Respect view **size**, **padding** and **gravity** properties when drawing to ensure your control content is properly sized and positioned within its rectangle
Custom Control - onDraw

- **drawARGB** - Fills canvas with color
- **drawRect** - Draws a rectangle using the provided Paint object for settings
- **drawOval, drawArc** - Draws ovals, arcs
- **drawPoint, drawPoints** - Draws points at the width in the provided Paint object
- **drawLine, drawLines, drawPath** - Draws lines and poly-lines using provided data
Custom Control - **onDraw**

- **drawBitmap, drawPicture** - Draws images using the provided source and destination rectangles for stretching.
- **drawText** - Renders text using the font information in the Paint object.
- **drawTextOnPath** - Places characters on the path, allowing for more interesting text placement.
Custom Control - **onDraw**

- **Canvas** also supports **advanced** drawing features
- **Clipping** using a clipping path
- **Matrix manipulation** of the drawing coordinate system
- **Stack of matrix and clip settings** accessible using the **save** and **restore** methods