Programming Handheld Systems
2D Graphics & Animation
Topics

2D Graphics
  ImageView
  Canvas

View Animation
Property Animation
Drawing 2D Graphics

Draw to a View

Simple graphics, little or no updating

Draw to a Canvas

More complex graphics, with regular updates
Drawable

Something that can be drawn, such as a bitmap, color, shape, etc.

Examples:

BitmapDrawable
ShapeDrawable
ColorDrawable
Drawing to Views

Can set Drawable objects on Views
Can do this via XML or programmatically
Graphics Bubble

Applications display a single ImageView. ImageView holds an image of a bubble.
ShapeDrawable

Used for drawing primitive shapes
Shape represented by a Shape class
PathShape - lines
RectShape - rectangles
OvalShape - ovals & rings
Applications display two Shapes within a RelativeLayout

The two shapes are partially overlapping and semi-transparent
Drawing with a Canvas

A Bitmap (a matrix of Pixels)
A Canvas for drawing to the underlying Bitmap
A Drawing Primitive (e.g. Rect, Path, Text, Bitmap)
A paint object (for setting drawing colors & styles)
Drawing Primitives

Canvas supports multiple drawing methods:

drawText()
drawPoints()
drawColor()
drawOval()
drawBitmap()
Paint

Specifies style parameters for drawing, e.g.,

setStrokeWidth()
setTextSize()
setColor()
setAntiAlias()
Application draws several boxes holding text, so using different paint settings each time.
Drawing with a Canvas

Can draw to generic Views, or to SurfaceViews
Drawing to Views

Use when updates are infrequent
Create a custom View class
System provides the canvas to the View when it calls the View’s onDraw() method
Create a Custom SurfaceView
Provide secondary thread for drawing
Application provides its own canvas and has greater control over drawing
GraphicsBubble

This application draws to custom View.
It has an Internal Thread that periodically wakes up and causes the View to move and to be redrawn.
Canvas with SurfaceView

Used for more high-performance drawing outside the UI thread
**SurfaceView**

*SurfaceView* manages a low-level drawing area called a *Surface*. The *Surface* represents a drawing area within the *View* hierarchy.
Defining a Custom SurfaceView

Subclass SurfaceView & implement SurfaceHolder.Callback

SurfaceHolder.Callback declares lifecycle methods that are called when the Surface changes
Using a SurfaceView

Set up SurfaceView

Draw to SurfaceView
Setup

Use SurfaceView's getHolder() to acquire Surface
Setup

Register for callbacks with SurfaceHolder's addCallback()
surfaceCreate()
surfaceChanged()
surfaceDestroyed()
Setup

Create the thread on which drawing operations will execute
Drawing

Acquire lock on Canvas

SurfaceHolder.lockCanvas()

Draw

Canvas.drawBitmap()

Unlock Canvas

SurfaceHolder.unlockCanvasAndPost()
View Animation

Changing the properties of a View over a period of time

Size

Position

Transparency

Orientation
View Animation Classes

TransitionDrawable
AnimationDrawable
Animation
TransitionDrawable

A 2-layer Drawable
Can fade between 1\textsuperscript{st} & 2\textsuperscript{nd} layers
GraphicsTransitionDrawable

This application uses the same shapes as the GraphicsShapeDraw applications.

Shows Cyan shape then fades to Magenta shape.
AnimationDrawable

Animates a series of Drawables
Each Drawable is shown for a specific amount of time
GraphicsFrameAnimation

Uses an Animation Drawable to present a frame by frame animation
Animation

A series of transformations applied to the content of a View
Can manipulate animation timing to give effect of sequential or simultaneous changes
GraphicsTweenAnimation

Application displays a single ImageView and animates several of its properties.
Property Animation

Animation – Changing properties of an Object over a period of time
Property Animation Architecture

ValueAnimator - Timing engine

TimeInterpolator - defines how values change as a function of time

AnimatorUpdateListener - called back at every animation frame change

TypeEvaluator - Calculates a property's value at a given point in time
Property Animation Architecture

AnimatorSet - combines individual animations to create more complex animations
GraphicsValueAnimator

Uses a ValueAnimator to animate changing an ImageView's background color
Same as the GraphicsTweenAnimation, Uses the ViewPropertyAnimator class, which is a simplified animator for Views
Next Time

MultiTouch & Gestures