



media  
computing  
group



**RWTH**AACHEN  
UNIVERSITY

# L13: Android Drawing and Animation

---

*Jonathan Diehl (Informatik 10)*  
*Hendrik Thüs (Informatik 9)*

# Drawing

# View

- Basic building block for user interface components
- Responsible for drawing and event handling
- Base class for widgets

# Custom View

Drawing	onDraw(Canvas)
Event processing	onKeyDown(int, KeyEvent)
	onKeyUp(int, KeyEvent)
	onTrackballEvent(MotionEvent)
	onTouchEvent(MotionEvent)

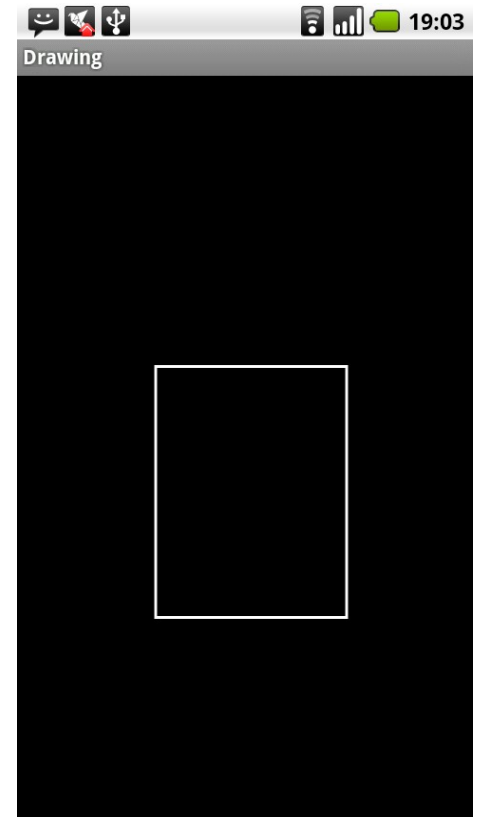
# Custom View Example

```
@Override
public boolean onTouchEvent(MotionEvent event) {
    int action = event.getAction();
    if (action == MotionEvent.ACTION_MOVE) {
        float x = event.getX();
        float y = event.getY();

        width = x - initX;
        height = y - initY;
    } else if (action == MotionEvent.ACTION_DOWN) {
        initX = event.getX();
        initY = event.getY();
        height = 1;
        width = 1;
    }
    invalidate();
    return true;
}
```

# Custom View Example (cont.)

```
private void init() {  
    paint.setStyle(Paint.Style.STROKE);  
    paint.setStrokeWidth(3);  
    paint.setColor(Color.WHITE);  
}  
  
@Override  
protected void onDraw(Canvas canvas) {  
    canvas.drawRect(initX + width, initY, initX,  
        initY + height, paint);  
}
```



# GestureDetector

# GestureDetector

- Detecting various gestures
- Uses MotionEvent
  - Used to report movement events
- OnGestureListener notifies when motion events occur
- For a subset of gestures:  
*GestureDetector.SimpleOnGestureListener*



# GestureDetector.OnGestureListener

abstract boolean	onDown(MotionEvent e)
abstract boolean	onFling(MotionEvent e1, MotionEvent e2, float velocityX, float velocityY)
abstract void	onLongPress(MotionEvent e)
abstract boolean	onScroll(MotionEvent e1, MotionEvent e2, float distanceX, float distanceY)
abstract void	onShowPress(MotionEvent e)
abstract boolean	onSingleTapUp(MotionEvent e)

# Demo



# Animations

# AnimationUtils

- Common utilities for working with animations

static Animation loadAnimation(Context context, int id)

Loads an Animation object from a resource

- Resource *android.R.anim*:
  - *fade\_in*, *fade\_out*, *slide\_in\_left*, *slide\_in\_right*

# Custom Animation

- Animation: abstract class
- Applicable to views, images, ...

protected void **applyTransformation** (float interpolatedTime, Transformation t)

public void **initialize** (int width, int height, int parentWidth, int parentHeight)

**width**                Width of the object being animated

**height**              Height of the object being animated

**parentWidth**        Width of the animated object's parent

**parentHeight**      Height of the animated object's parent

# Animation Example

```
public class MyAnimation extends Animation {
    @Override
    public void initialize(int width, int height, int parentWidth,
        int parentHeight) {
        super.initialize(width, height, parentWidth, parentHeight);
        setDuration(5000);
        setFillAfter(true);
        setInterpolator(new LinearInterpolator());
    }

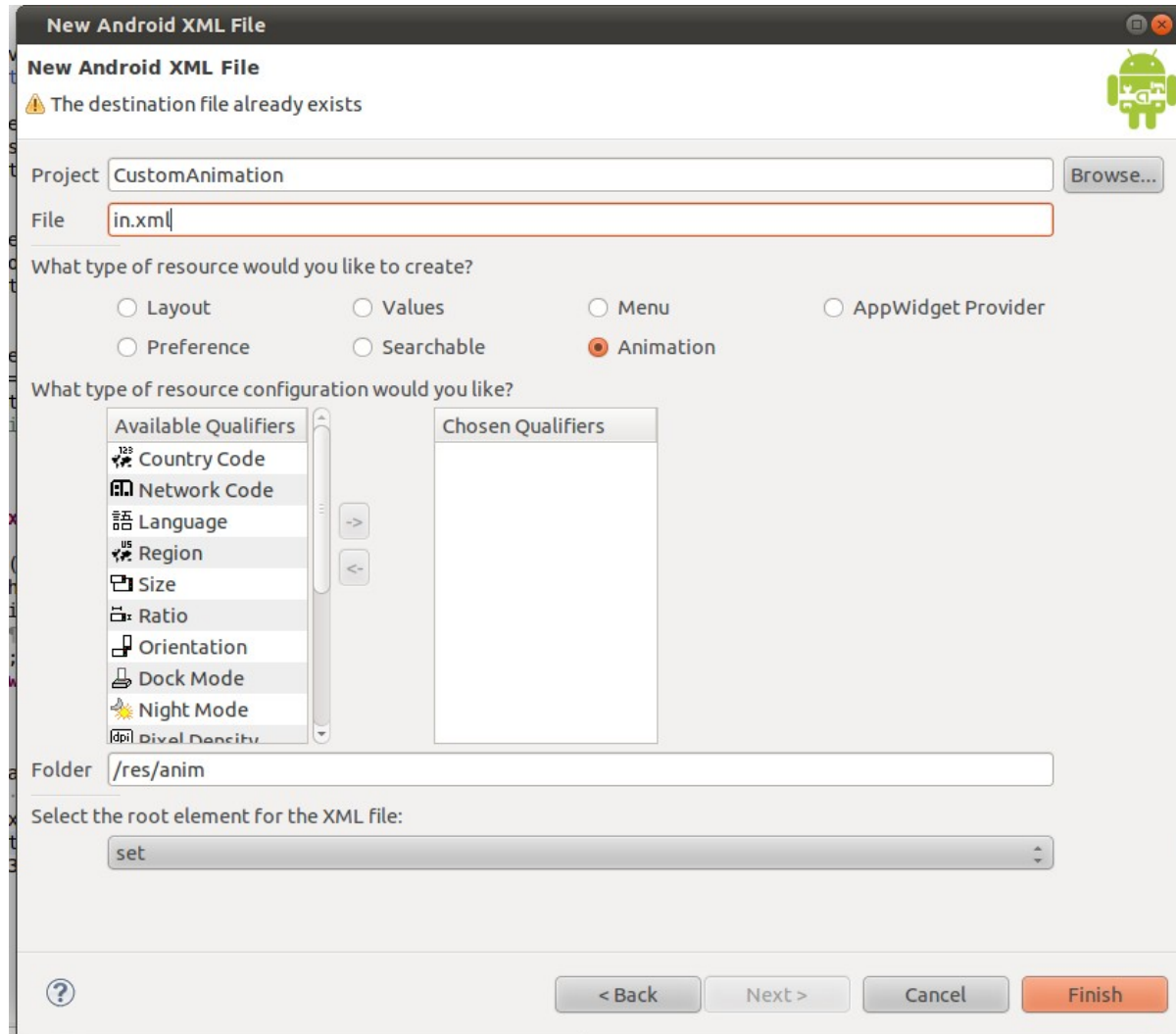
    @Override
    protected void applyTransformation(float interpolatedTime,
        Transformation t) {
        final Matrix matrix = t.getMatrix();
        matrix.setScale(interpolatedTime, interpolatedTime);
        matrix.postRotate(360 * interpolatedTime);
    }
}
```

# Animation Resource

- Tween animation
- Defined in XML
- Transitions
  - Rotation
  - Fading
  - Moving
  - Stretching

<http://developer.android.com/guide/topics/resources/animation-resource.html#Tween>

# Animation Resource





# Animation Resource

```
<?xml version="1.0" encoding="utf-8"?>  
<set xmlns:android="http://schemas.android.com/apk/res/android">  
  <translate android:fromYDelta="100%" android:toYDelta="0"  
    android:duration="3000" />  
  <alpha android:fromAlpha="0.0" android:toAlpha="1.0"  
    android:duration="3000" />  
</set>
```

# Demo

