

CS 646 Android Mobile Application Development
Spring Semester, 2015
Doc 16 Testing & Some Tools
Apr 6, 2015

Copyright ©, All rights reserved. 2015 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/openpub/>) license defines the copyright on this document.

Testing References

JUnit Cookbook <http://junit.sourceforge.net/doc/cookbook/cookbook.htm>

JUnit Test Infected: Programmers Love Writing Tests <http://junit.sourceforge.net/doc/testinfected/testing.htm>

JUnit Javadoc: <http://www.junit.org/junit/javadoc/3.8/index.htm>, <http://junit.org/junit/javadoc/4.5/>

JUnit FAQ, <http://junit.sourceforge.net/doc/faq/faq.htm>

Testing for Programmers, Brian Marick, Available at: <http://www.exampler.com/testing-com/writings.html>

Android Documentation, <http://developer.android.com/tools/testing/index.html>

Future Topics

Testing and Tools

Back End as Service (BaaS)

Web

Maps

Location

Templates

Styles & Drawing

Services

Notifications

Testing

Testing

Johnson's Law

If it is not tested it does not work

The more time between coding and testing

- More effort is needed to write tests

- More effort is needed to find bugs

- Fewer bugs are found

- Time is wasted working with buggy code

- Development time increases

- Quality decreases

Unit Testing

Tests individual code segments

Automated tests

XUnit

Free frameworks for Unit testing

SUnit originally written by Kent Beck 1994

JUnit written by Kent Beck & Erich Gamma

Available at: <http://www.junit.org/>

Ports to many languages at:

<http://www.xprogramming.com/software.htm>

Android and JUnit

Support of JUnit part of Android

Only supports JUnit 3 type tests

JUnit Example - JUnit 3.x

Goal: Implement a Stack containing integers.

Tests:

- Subclass `junit.framework.TestCase`

- Methods starting with 'test' are run by `TestRunner`

Sample Testcase

```
import junit.framework.*;
public class Sampletest extends TestCase {
    public void testPushPop() {
        Stack<String> test = new Stack<String>();
        assertTrue( test.isEmpty() );
        test.push("A");
        assertFalse( test.isEmpty() );
        test.push("B");
        test.push("C");
        assertEquals("C", test.pop());
        assertEquals("B", test.pop());
        assertEquals("A", test.pop());
        assertTrue( test.isEmpty() );
        try {
            test.pop();
            fail();
        } catch (EmptyStackException e) {
        }
    }
}
```

Assert Methods

assertTrue()
assertFalse()
assertEquals()
assertNotEquals()
assertSame()
assertNotSame()
assertNull()
assertNotNull()
fail()

For a complete list see

<http://junit.sourceforge.net/javadoc/org/junit/Assert.html>

Android Testing

Unit Testing

Application logic independent of UI/OS events

Normal JUnit tests

Logic dependent on UI/OS events

Require special environment

Some Android Specific Test Classes

ActivityInstrumentationTestCase2

- Functional Tests of multiple activities
- Can click on buttons, enter text in fields etc
- Can test lifecycle methods

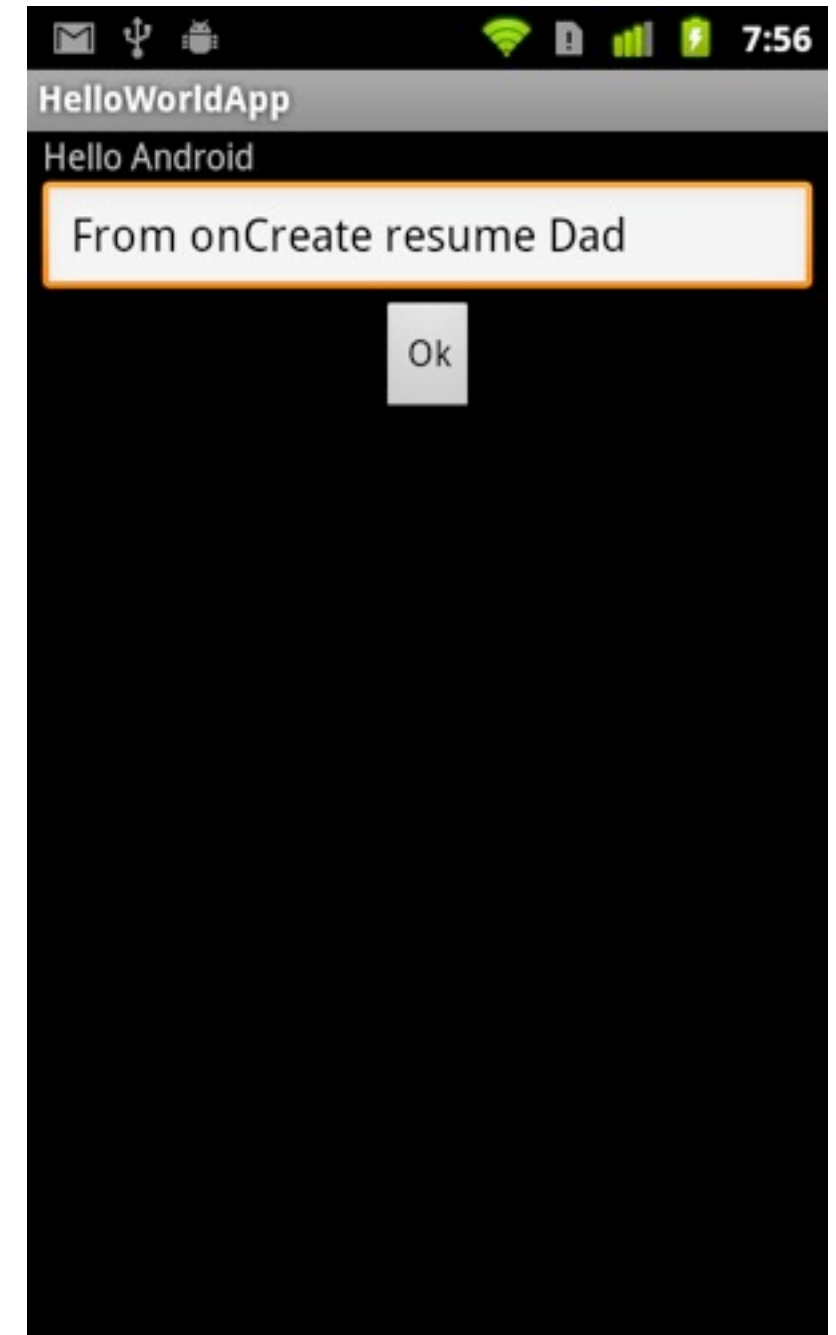
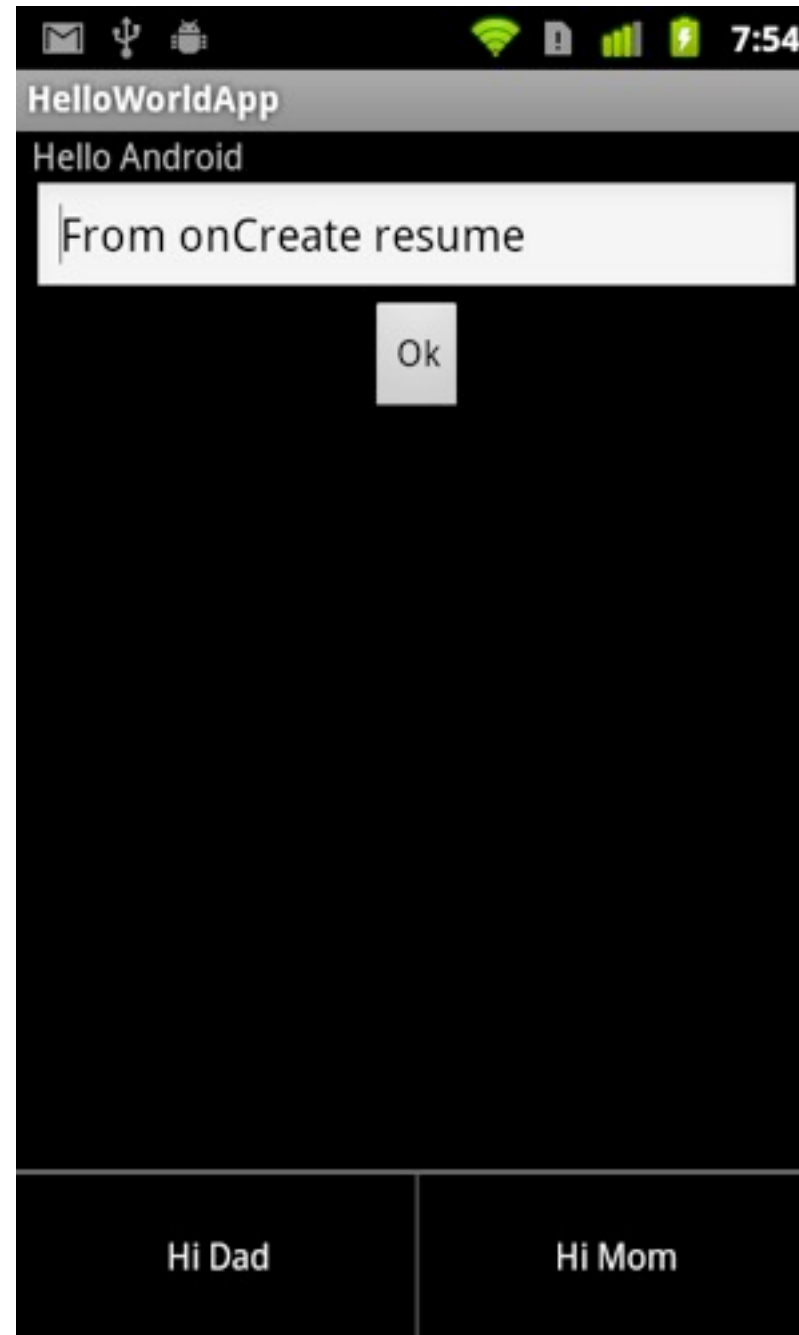
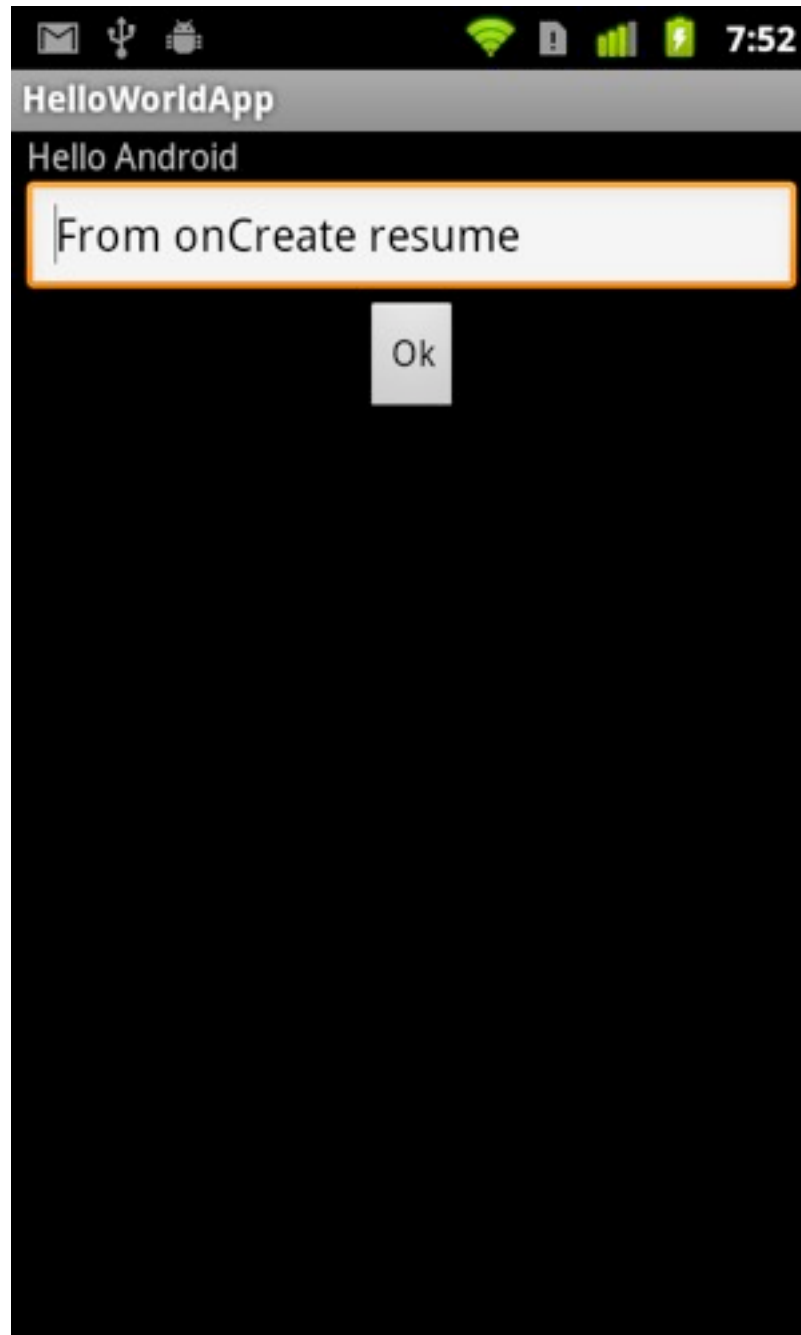
ActivityUnitTestCase

- Can run Android activities
- Can just call methods on your class

ApplicationTestCase

- Test the setup of Application

Application to test



Application to test

```
package edu.sdsu.cs.whitney;
```

```
//imports not listed
```

```
public class TestingExampleActivity extends Activity implements View.OnClickListener {  
    private EditText messageText;  
    private static final int DAD_ID = Menu.FIRST;  
    private static final int MOM_ID = Menu.FIRST + 1;  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
        messageText = (EditText) this.findViewById(R.id.message);  
        messageText.setText("From onCreate");  
        Button ok = (Button) findViewById(R.id.ok);  
        ok.setOnClickListener(this);  
    }  
}
```


Application to test

```
public void onClick(View v) {  
    messageText.setText(messageText.getText() + " click");  
}
```

```
protected void onPause() {  
    messageText.setText(messageText.getText() + " pause");  
    super.onPause();  
}
```

```
protected void onResume() {  
    super.onResume();  
    messageText.setText(messageText.getText() + " resume");  
}
```

Application to test

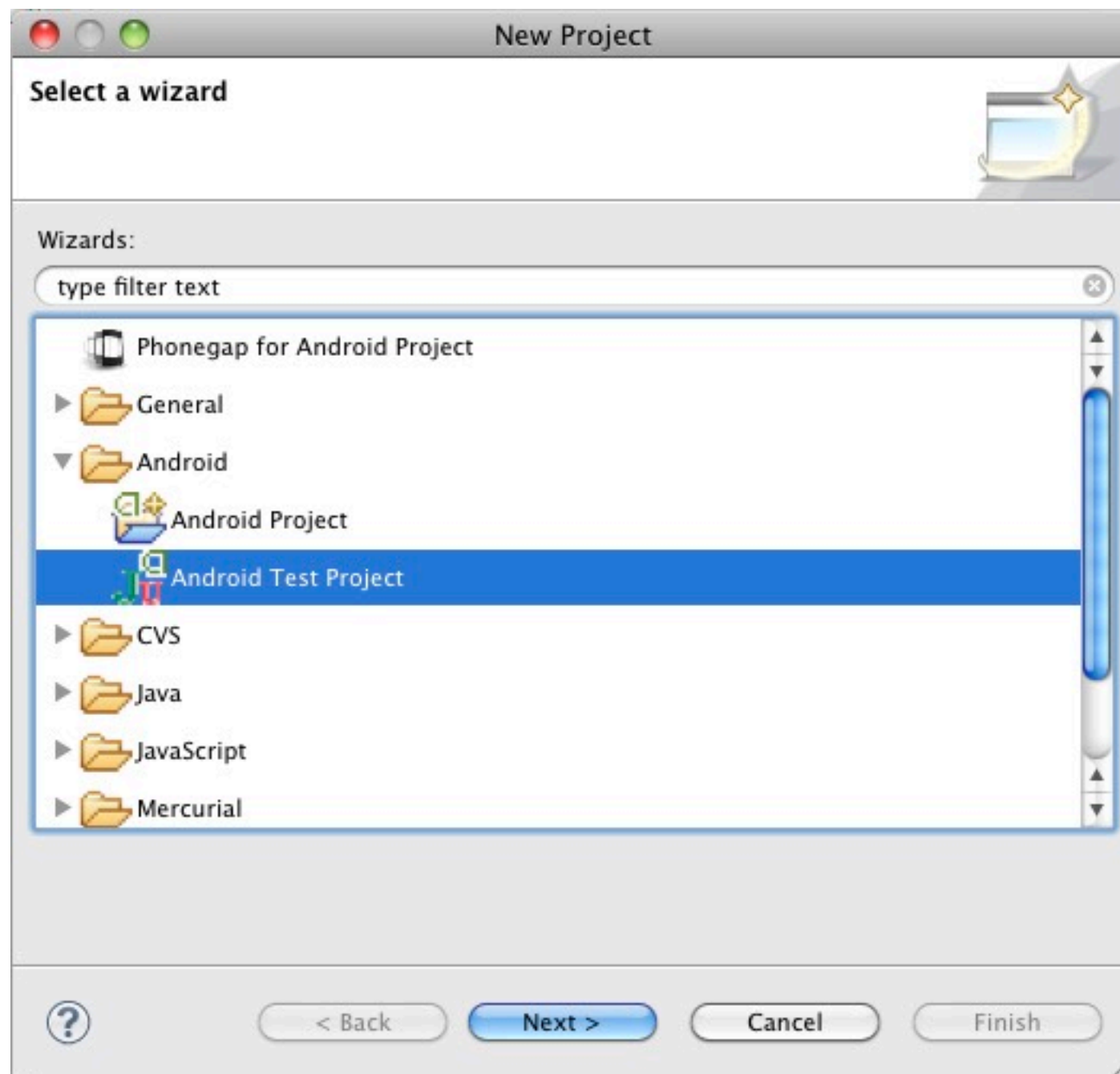
```
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);

    menu.add(0, DAD_ID, 0, R.string.menu_dad).setShortcut('0', 'd');
    menu.add(0, MOM_ID, 0, R.string.menu_mom);
    return true;
}

public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case DAD_ID:
            messageText.setText(messageText.getText() + " Dad");
            return true;
        case MOM_ID:
            messageText.setText(messageText.getText() + " Mom");
            return true;
    }
    return super.onOptionsItemSelected(item);
}
```

Test Setup

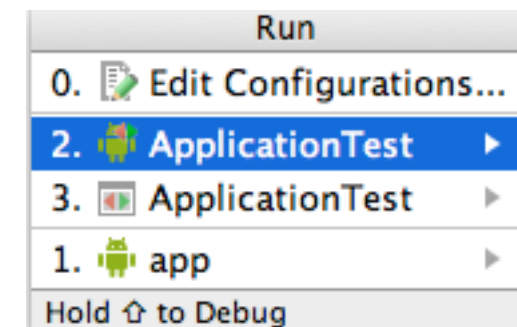
http://developer.android.com/resources/tutorials/testing/helloandroid_test.html



Android Studio

Adds tests when create projects

But uses ApplicationTestCase
So need to change



Running Tests in Android Studio



After you run the test the first time “Run” will run the tests not the app

Start of test

```
package edu.sdsu.cs.whitney.test;

// imports not shown

public class InstrumenetedTests extends
    ActivityInstrumentationTestCase2<TestingExampleActivity> {
    private HelloAndroid mActivity;
    private EditText mView;
    private Button mButton;
    private Instrumentation mInstrumentation;

    public HelloAndroidTest() {
        super("edu.sdsu.cs.whitney", TestingExampleActivity.class);
    }
}
```

Set up

```
private String getText() {  
    String resultText = mView.getText().toString();  
    Log.i("rew", resultText);  
    return resultText;  
}  
  
protected void setUp() throws Exception {  
    super.setUp();  
    mInstrumentation = getInstrumentation();  
    mActivity = this.getActivity();  
    mView = (EditText) mActivity.findViewById(sdsu.cs696.R.id.message);  
    mButton = (Button) mActivity.findViewById(sdsu.cs696.R.id.ok);  
}
```

Tests

```
public void testPreconditions() {  
    assertNotNull(mView);  
    assertNotNull(mButton);  
    assertEquals(getText(), "From onCreate resume");  
}
```

```
public void testButton() {  
    mActivity.runOnUiThread(new Runnable() {  
        public void run() {  
            mButton.performClick();  
        }  
    });  
    mInstrumentation.waitForIdleSync();  
    assertEquals(getText(), "From onCreate resume click");  
}
```


Tests

```
public void testPause() {  
    mActivity.runOnUiThread(new Runnable() {  
        public void run() {  
            mInstrumentation.callActivityOnPause(mActivity);  
        }  
    });  
    mInstrumentation.waitForIdleSync();  
    assertEquals(getText(), "From onCreate resume pause");  
}
```

```
public void testMenu() {  
  
    final boolean didMenuRun = mInstrumentation.invokeMenuActionSync(  
        mActivity, Menu.FIRST, 1);  
    assertTrue(didMenuRun);  
  
    assertEquals(getText(), "From onCreate resume Dad");  
}
```

Testing LifeCycle Events

Examples use Assignment 1 code

First activity called HomeEnterURL

URL field called url_text

The Test Class

```
public class HomeEnterURLLifeCycle extends
    ActivityInstrumentationTestCase2<HomeEnterURL> {
    private HomeEnterURL mActivity;

    public HomeEnterURLLifeCycle() {
        super("cs646.AndroidAssignment1", HomeEnterURL.class);
    }

    protected void setUp() throws Exception {
        super.setUp();
        mActivity = getActivity();
    }
}
```

Pause

```
@UiThreadTest
public void testStatePause() {
    Instrumentation instrument = this.getInstrumentation();
    EditText urlText = (EditText) mActivity.findViewById(R.id.url_text);
    urlText.setText("www.sdsu.edu");
    instrument.callActivityOnPause(mActivity);
    urlText.setText("foo");
    instrument.callActivityOnResume(mActivity);
    assertEquals("www.sdsu.edu", urlText.getText().toString());
}
```

Destroy

```
@UiThreadTest
public void testStateDestroy() {
    Instrumentation instrument = this.getInstrumentation();
    EditText urlText = (EditText) mActivity.findViewById(R.id.url_text);
    urlText.setText("www.sdsu.edu");
    mActivity.finish();
    mActivity = getActivity();
    instrument.callActivityOnResume(mActivity);
    urlText = (EditText) mActivity.findViewById(R.id.url_text);
    assertEquals("www.sdsu.edu", urlText.getText().toString());
}
```

To learn more

See Android testing Tutorial at:

<http://developer.android.com/tools/testing/index.html>

Monkey Testing

Monkey

Generates random events for your activity

Enters text

Click buttons

Selects menus

Rotates screen

etc.

Sample Run

Install app on emulator or device

adb is in the platform-tools directory in Android installation

You give package name of application

```
Al pro 23->adb shell monkey -p edu.sdsu.cs.whitney.testingexample 500
```

```
Events injected: 500
```

```
## Network stats: elapsed time=7681ms (0ms mobile, 7681ms wifi, 0ms not connected)
```

-p list the package of your app

Verbose Mode

Al pro 24->adb shell monkey -p edu.sdsu.cs.whitney.testingexample -v 10

:Monkey: seed=0 count=10

:AllowPackage: edu.sdsu.cs.whitney.testingexample

:IncludeCategory: android.intent.category.LAUNCHER

:IncludeCategory: android.intent.category.MONKEY

// Event percentages:

// 0: 15.0%

// 1: 10.0%

// 2: 15.0%

// 3: 25.0%

// 4: 15.0%

// 5: 2.0%

// 6: 2.0%

// 7: 1.0%

// 8: 15.0%

:Switch:

#Intent;action=android.intent.action.MAIN;category=android.intent.category.LAUNCHER;

Flags=0x10000000;component=sdsu.cs696/.HelloAndroid;end

// Allowing start of Intent { act=android.intent.action.MAIN
cat=[android.intent.category.LAUNCHER] cmp=sdsu.cs696/.HelloAndroid } in package
sdsu.cs696

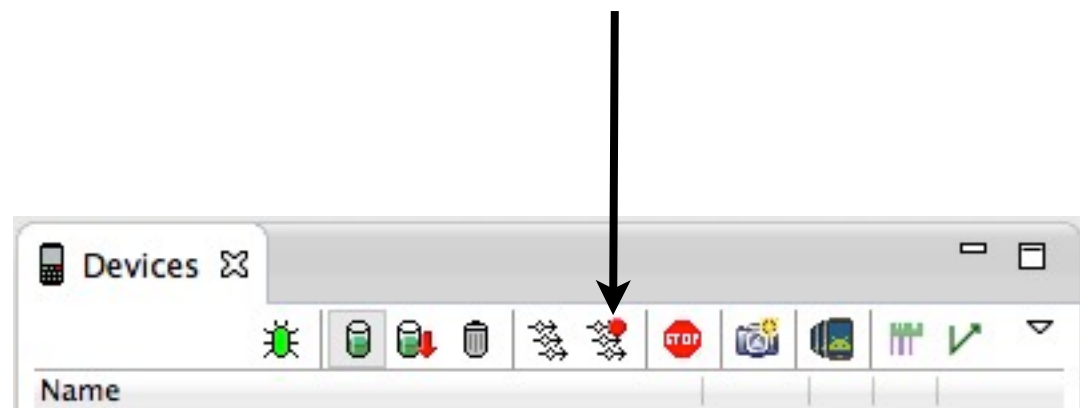
Performance Measuring

Profiling Time Usage

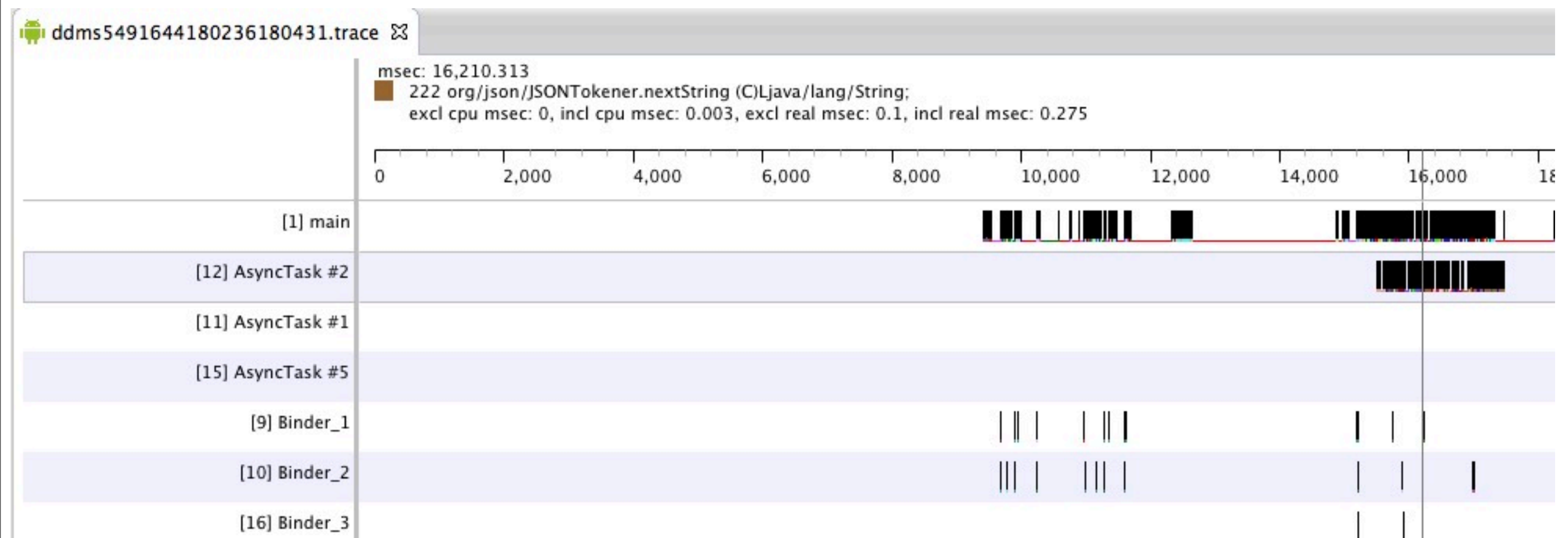
DDMS View - Eclipse for Android

Android Device Monitor - Android Studio

Start/Stop profiling



Trace View



Method Call View

Method Name	Incl Cpu Time %	Incl Cpu Time	Excl Cpu Time %	Excl Cpu Time	Incl Real Time %	Incl Real Time	Excl Real Time
0 (toplevel)	100.0%	4799.372	0.3%	14.185	100.0%	69337.065	
1 android/os/Handler.dispatchMessage (Landroid/os/Message;)V	73.7%	3536.392	0.1%	2.594	8.9%	6161.389	
2 android/os/Handler.handleCallback (Landroid/os/Message;)V	52.6%	2524.468	0.0%	1.470	6.8%	4697.739	
3 android/view/Choreographer\$FrameDisplayEventReceiver.run ()V	51.1%	2452.062	0.0%	1.436	6.4%	4425.352	
Parents							
Children							
self	0.1%	1.436			0.0%	1.043	
4 android/view/Choreographer.doFrame (J)V	99.9%	2450.626			100.0%	4423.667	
(context switch)	0.0%	0.000			0.0%	0.642	
4 android/view/Choreographer.doFrame (J)V	51.1%	2450.626	0.2%	7.490	6.4%	4423.667	
5 android/view/Choreographer.doCallbacks (J)V	50.9%	2441.896	0.2%	8.906	6.4%	4416.664	
6 android/view/Choreographer\$CallbackRecord.run ()V	50.5%	2425.721	0.1%	4.377	6.3%	4398.599	
7 android/view/ViewRootImpl\$TraversalRunnable.run ()V	48.7%	2337.119	0.1%	3.115	6.2%	4267.576	
8 android/view/ViewRootImpl.doTraversal ()V	48.6%	2334.004	0.1%	5.866	6.2%	4266.667	
9 android/view/ViewRootImpl.performTraversals ()V	48.3%	2316.307	0.3%	12.857	6.1%	4249.688	
10 android/view/ViewRootImpl.performDraw ()V	22.6%	1084.767	0.1%	2.862	3.6%	2518.160	

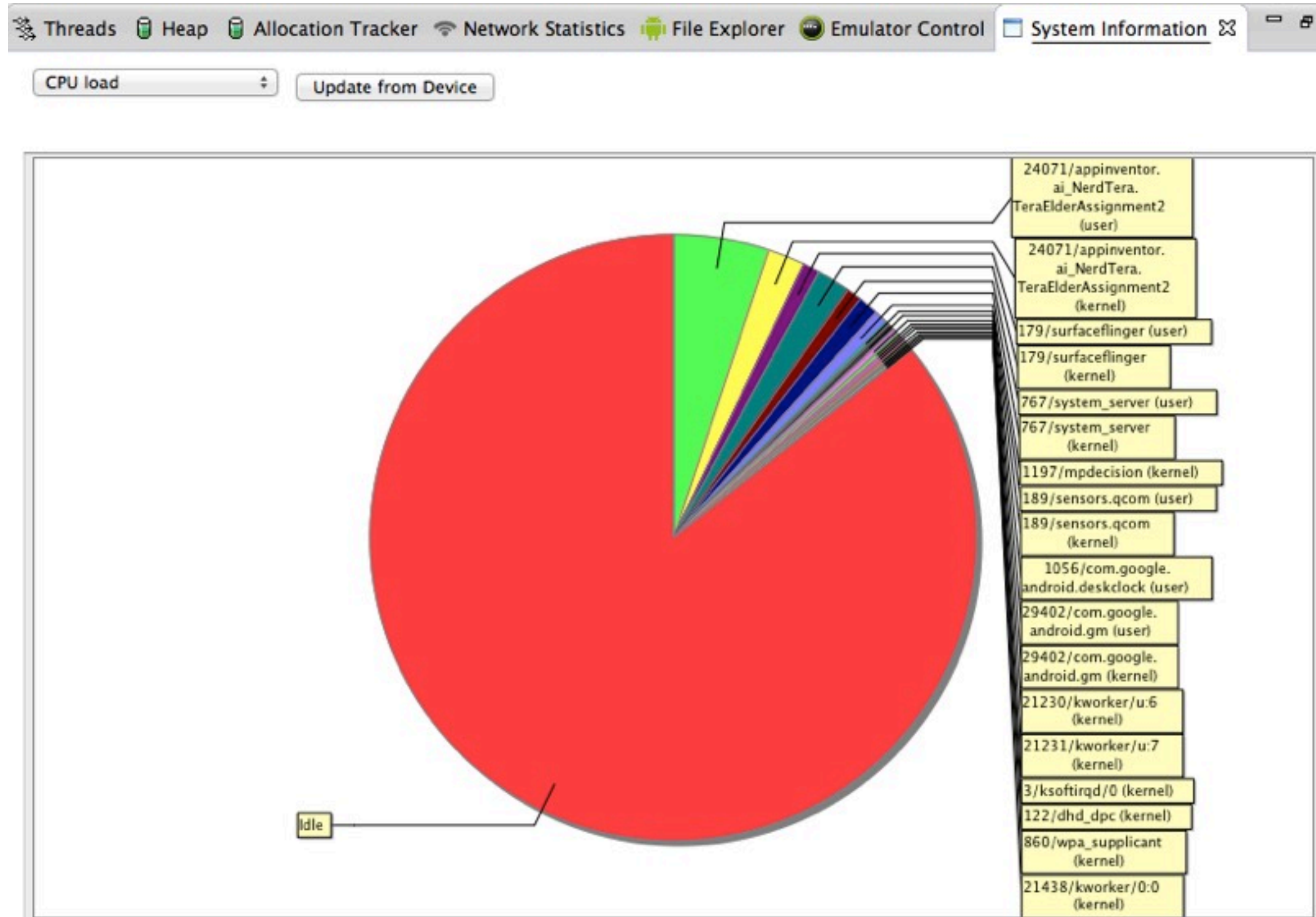
Incl CPU Time - time spent in the function including functions it calls

Excl CPU Time - time spent in the function excluding time spent in functions it calls

Real (Wall) time - Includes all time from start of function to end of function

CPU time - does not include I/O time or swap time

System Information - CPU Load



System Information - Frame Render Time



Draw -

Java code drawing

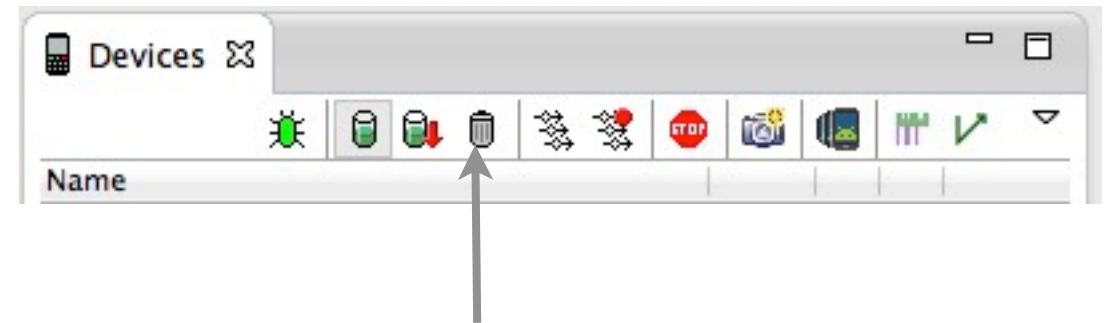
Process

Android 2D renderer

Execute

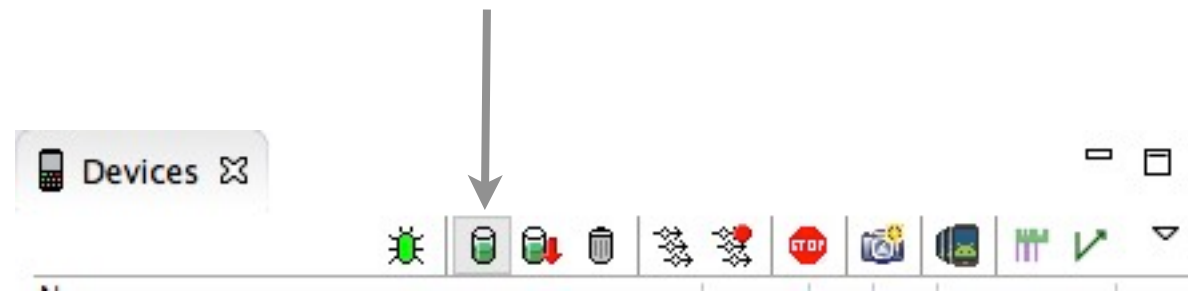
Send Frame to
Composer

Garbage Collection Output



Reason	Amount Freed	% of Heap Free	Number of Objects/ Heap Size	Timed Paused
GC_FOR_ALLOC freed <1K, 19% free 7491K/9208K, paused 5ms, total 6ms				
GC_FOR_ALLOC freed 1285K, 32% free 6285K/9208K, paused 8ms, total 18ms				
GC_FOR_ALLOC freed 396K, 39% free 7011K/11396K, paused 28ms, total 32ms				
GC_FOR_ALLOC freed 752K, 42% free 6630K/11396K, paused 49ms, total 50ms				
GC_FOR_ALLOC freed 384K, 39% free 7005K/11396K, paused 9ms, total 10ms				
GC_FOR_ALLOC freed 749K, 42% free 6628K/11396K, paused 26ms, total 26ms				
GC_EXPLICIT freed 875K, 29% free 6616K/9208K, paused 2ms+0ms, total 21ms				

Heap



Threads Heap Allocation Tracker Network Statistics File Explorer Emulator C

Heap updates will happen after every GC for this client

ID	Heap Size	Allocated	Free	% Used	# Objects
1	31.898 MB	29.099 MB	2.799 MB	91.22%	51,241

Cause GC

Display: Stats

Type	Count	Total Size	Smallest	Largest	Median	Average
free	1,049	2.717 MB	16 B	1.831 MB	32 B	2.651 KB
data object	31,452	1.072 MB	16 B	1.062 KB	32 B	35 B
class object	3,722	1.000 MB	168 B	42.375 KB	168 B	281 B
1-byte array (byte[], boolean[])	537	25.916 MB	24 B	7.910 MB	600 B	49.419 KB
2-byte array (short[], char[])	10,609	787.859 KB	24 B	76.227 KB	48 B	76 B
4-byte array (object[], int[], float[])	4,673	330.812 KB	24 B	16.023 KB	40 B	72 B
8-byte array (long[], double[])	248	18.625 KB	24 B	4.008 KB	40 B	76 B
non-Java object	1,426	69.602 KB	16 B	480 B	32 B	49 B

Heap - More Data

Threads Heap Allocation Tracker Network Statistics File Explorer Emulator C

Heap updates will happen after every GC for this client

ID	Heap Size	Allocated	Free	% Used	# Objects
1	31.898 MB	29.099 MB	2.799 MB	91.22%	51,241

Cause GC

Display: Stats

Type	Count	Total Size	Smallest	Largest	Median	Average
free	1,049	2.717 MB	16 B	1.831 MB	32 B	2.651 KB
data object	31,452	1.072 MB	16 B	1.062 KB	32 B	35 B
class object	3,722	1.000 MB	168 B	42.375 KB	168 B	281 B
1-byte array (byte[], boolean[])	537	25.916 MB	24 B	7.910 MB	600 B	49.419 KB
2-byte array (short[], char[])	10,609	787.859 KB	24 B	76.227 KB	48 B	76 B
4-byte array (object[], int[], float[])	4,673	330.812 KB	24 B	16.023 KB	40 B	72 B
8-byte array (long[], double[])	248	18.625 KB	24 B	4.008 KB	40 B	76 B
non-Java object	1,426	69.602 KB	16 B	480 B	32 B	49 B

Allocation Tracker

Threads
Heap
Allocation Tracker
Network Statistics
File Explorer
Emulator Control
System Information

Stop Tracking
Get Allocations
Filter:

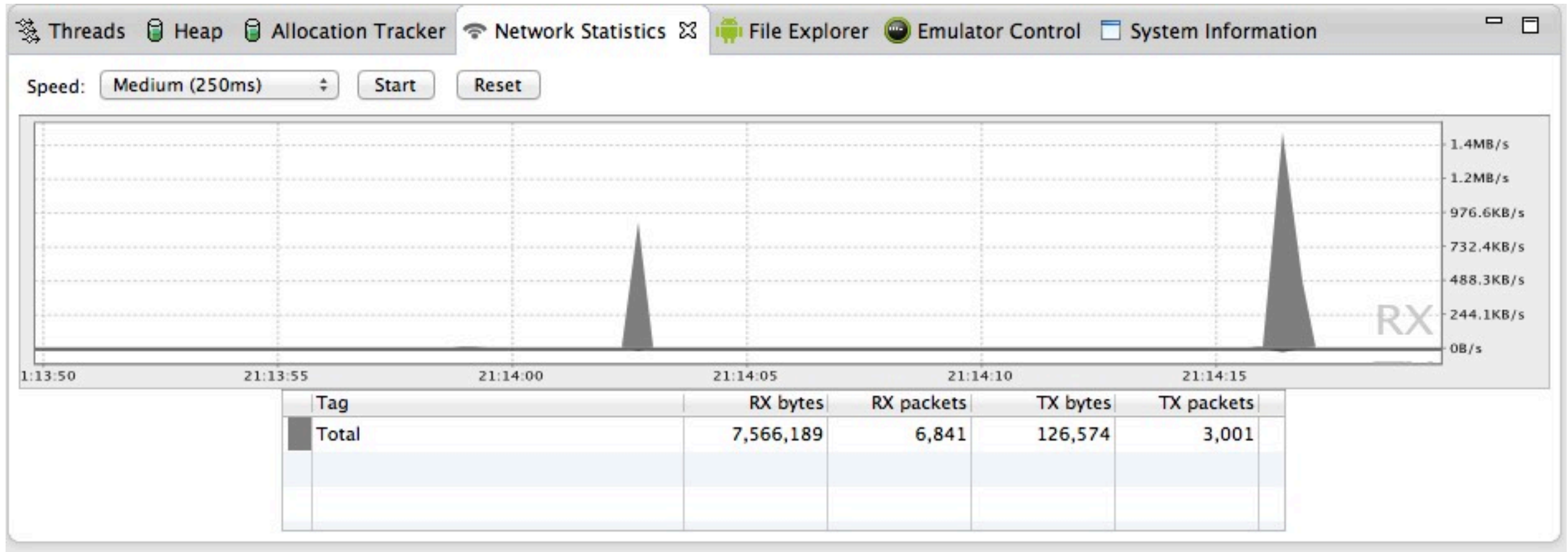
☐ Inc.

Alloc Order	Allocation Size	Allocated Class	Thread Id	Allocated in	Allocated in
17343	1920016	byte[]	1	android.graphics.BitmapFactory	nativeDecodeAsset
205	103768	byte[]	15	android.graphics.BitmapFactory	nativeDecodeStream
104	103768	byte[]	1	android.graphics.BitmapFactory	nativeDecodeStream
14470	103696	byte[]	11	android.graphics.BitmapFactory	nativeDecodeStream
14368	103696	byte[]	1	android.graphics.BitmapFactory	nativeDecodeStream
9533	103696	byte[]	14	android.graphics.BitmapFactory	nativeDecodeStream
9431	103696	byte[]	1	android.graphics.BitmapFactory	nativeDecodeStream
5213	103384	byte[]	16	android.graphics.BitmapFactory	nativeDecodeStream
5100	103384	byte[]	1	android.graphics.BitmapFactory	nativeDecodeStream
2674	103384	byte[]	15	android.graphics.BitmapFactory	nativeDecodeStream
2565	103384	byte[]	1	android.graphics.BitmapFactory	nativeDecodeStream
22986	82960	byte[]	1	android.graphics.BitmapFactory	nativeDecodeAsset
11993	77888	byte[]	14	android.graphics.BitmapFactory	nativeDecodeStream
11876	77888	byte[]	1	android.graphics.BitmapFactory	nativeDecodeStream

```

at android.graphics.BitmapFactory.nativeDecodeAsset(Native Method)
at android.graphics.BitmapFactory.decodeStream(BitmapFactory.java:587)
at android.graphics.BitmapFactory.decodeResourceStream(BitmapFactory.java:422)
at android.graphics.drawable.Drawable.createFromResourceStream(Drawable.java:840)
at android.content.res.Resources.loadDrawable(Resources.java:2110)
at android.content.res.Resources.getDrawable(Resources.java:700)
at android.widget.ImageView.resolveUri(ImageView.java:638)
at android.widget.ImageView.setImageResource(ImageView.java:367)
at edu.prouty.hw3.photogallery.ImageFragment.onCreateView(ImageFragment.java:81)
at android.support.v4.app.Fragment.performCreateView(Fragment.java:1460)
at android.support.v4.app.FragmentManagerImpl.moveToState(FragmentManager.java:911)
at android.support.v4.app.FragmentManagerImpl.moveToState(FragmentManager.java:1088)
at android.support.v4.app.BackStackRecord.run(BackStackRecord.java:682)
at android.support.v4.app.FragmentManagerImpl.execPendingActions(FragmentManager.java:1167)
at android.support.v4.app.FragmentManagerImpl.executePendingTransactions(FragmentManager.java:1392)
at android.support.v4.app.FragmentPagerAdapter.finishUpdate(FragmentPagerAdapter.java:143)
    
```

Network Statistics



RX - Received data

TX - Transmitted data

Interface Testing

Hierarchy Viewer

<http://developer.android.com/guide/developing/debugging/debugging-ui.html>

located in <sdk>/tools

Pixel Perfect Window

- View UI at pixel level

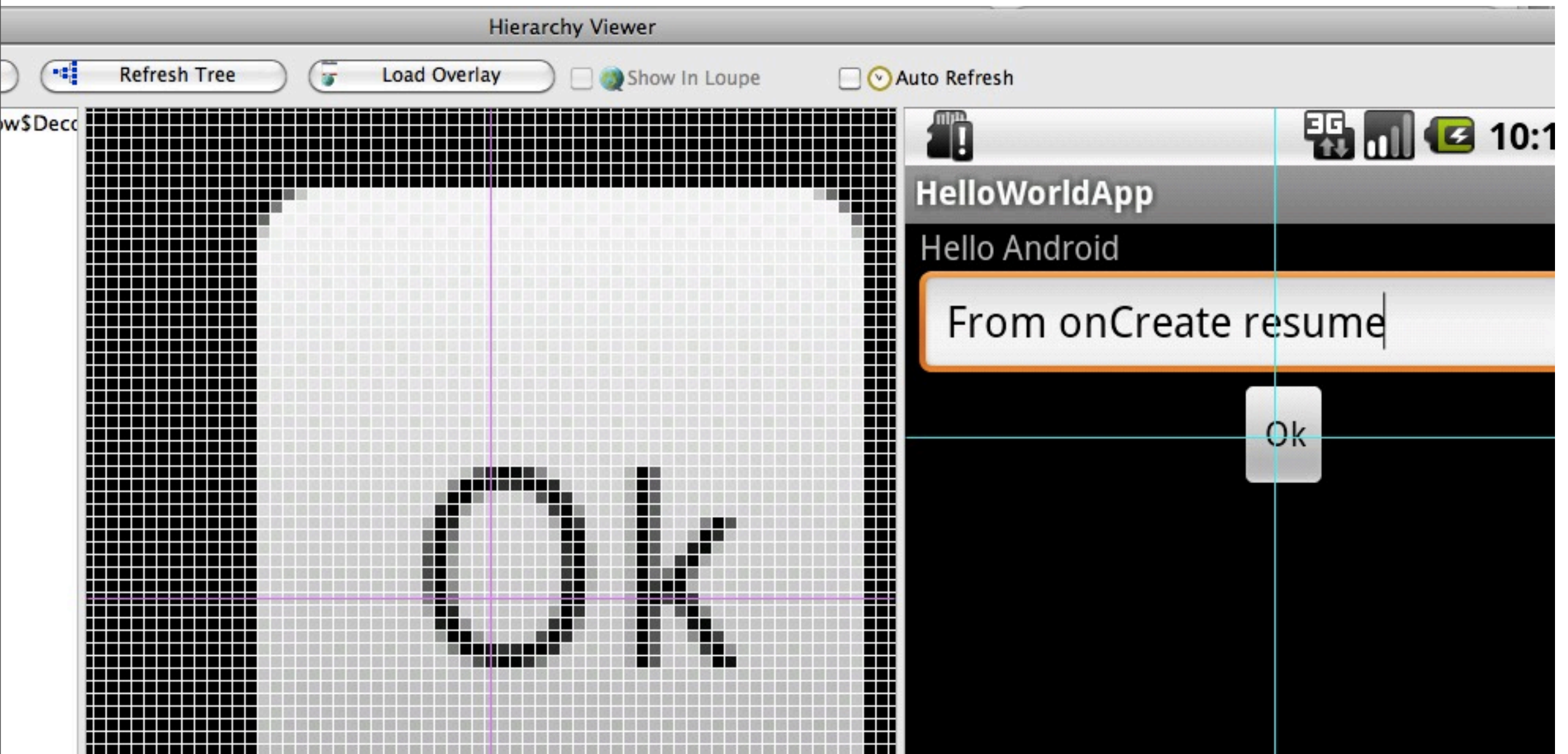
View Hierarchy Window

- View hierarchy structure of UI

- See all view properties

- Measure render time of each screen

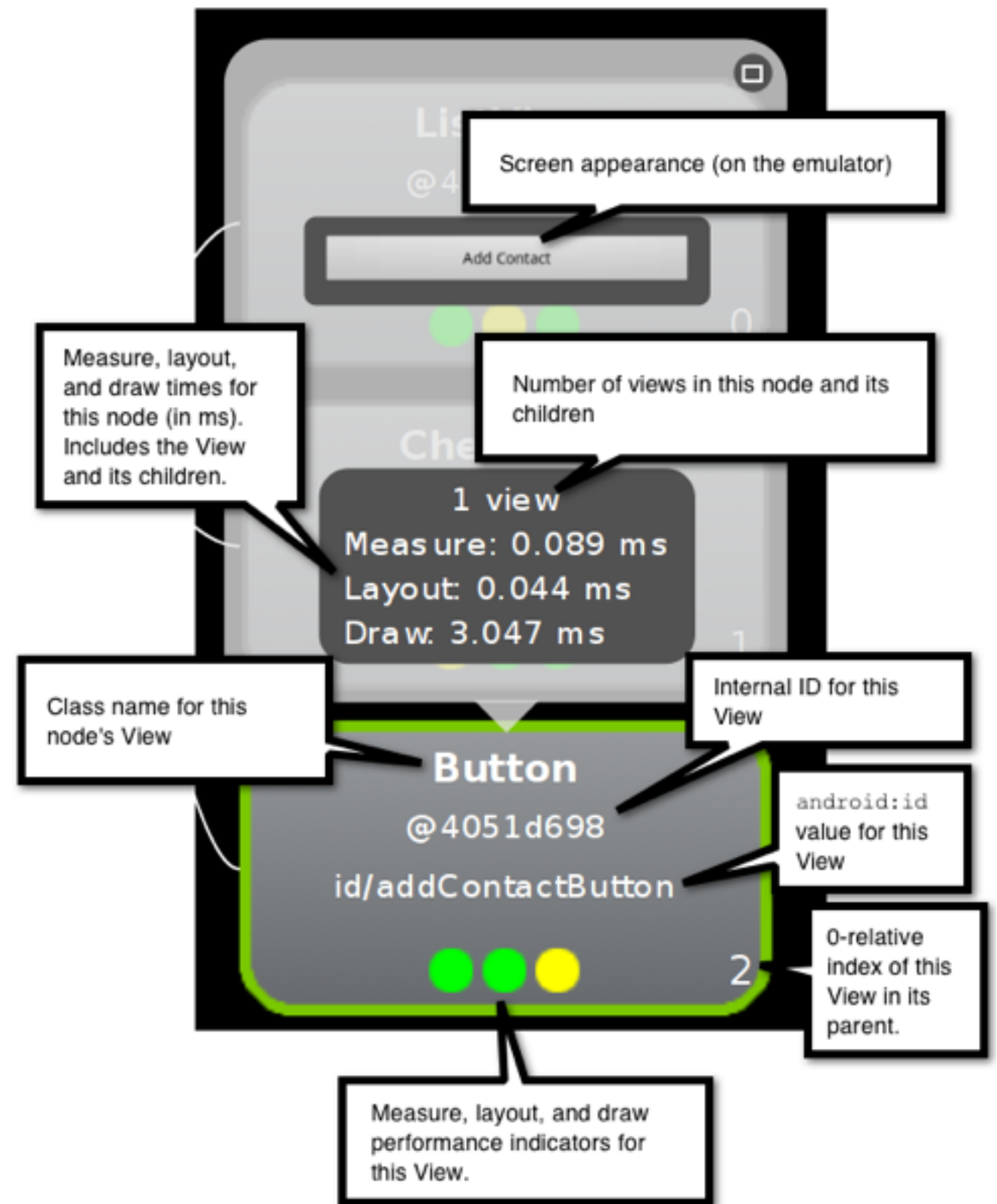
Pixel Perfect Window



View Hierarchy Window



Info Key



layoutopt

Finds inefficiencies in the view hierarchy in xml layout files

```
<sdk>/tools/layoutopt <xmlFiles>
```

Al pro 33->layoutopt /Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/*
/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/activity_animation.xml

28:28 Use an android:layout_height of 0dip instead of wrap_content for better performance

/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/alarm_controller.xml

28:28 Use an android:layout_height of 0dip instead of wrap_content for better performance

/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/alarm_service.xml

28:28 Use an android:layout_height of 0dip instead of wrap_content for better performance

/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/alert_dialog.xml

25:50 This LinearLayout tag should use android:layout_height="wrap_content"