

CS 696 Emerging Web and Mobile Technologies
Spring Semester, 2011
Doc 25 Android Graphs & Tabs
Apr 28, 2011

Copyright ©, All rights reserved. 2011 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

References

Android Tab tutorial

<http://developer.android.com/resources/tutorials/views/hello-tabwidget.html>

AndroidPlot Documentation, <http://androidplot.com/wiki/Docs>

Android Graphing libraries

Name	License Type	Price
AndroidPlot	Modified BSD	Free
Android Chart	Commercial	\$299-\$999
Chartdroid	Apache License 2.0	Free
GraphView	Custom	Mandatory Donation
RChart 2	Custom	\$138.60
AChartEngine	Apache License 2.0	Free
aiCharts	Commercial	\$299 and up

AndroidPlot

<http://androidplot.com/wiki/Home>

Version 0.4a

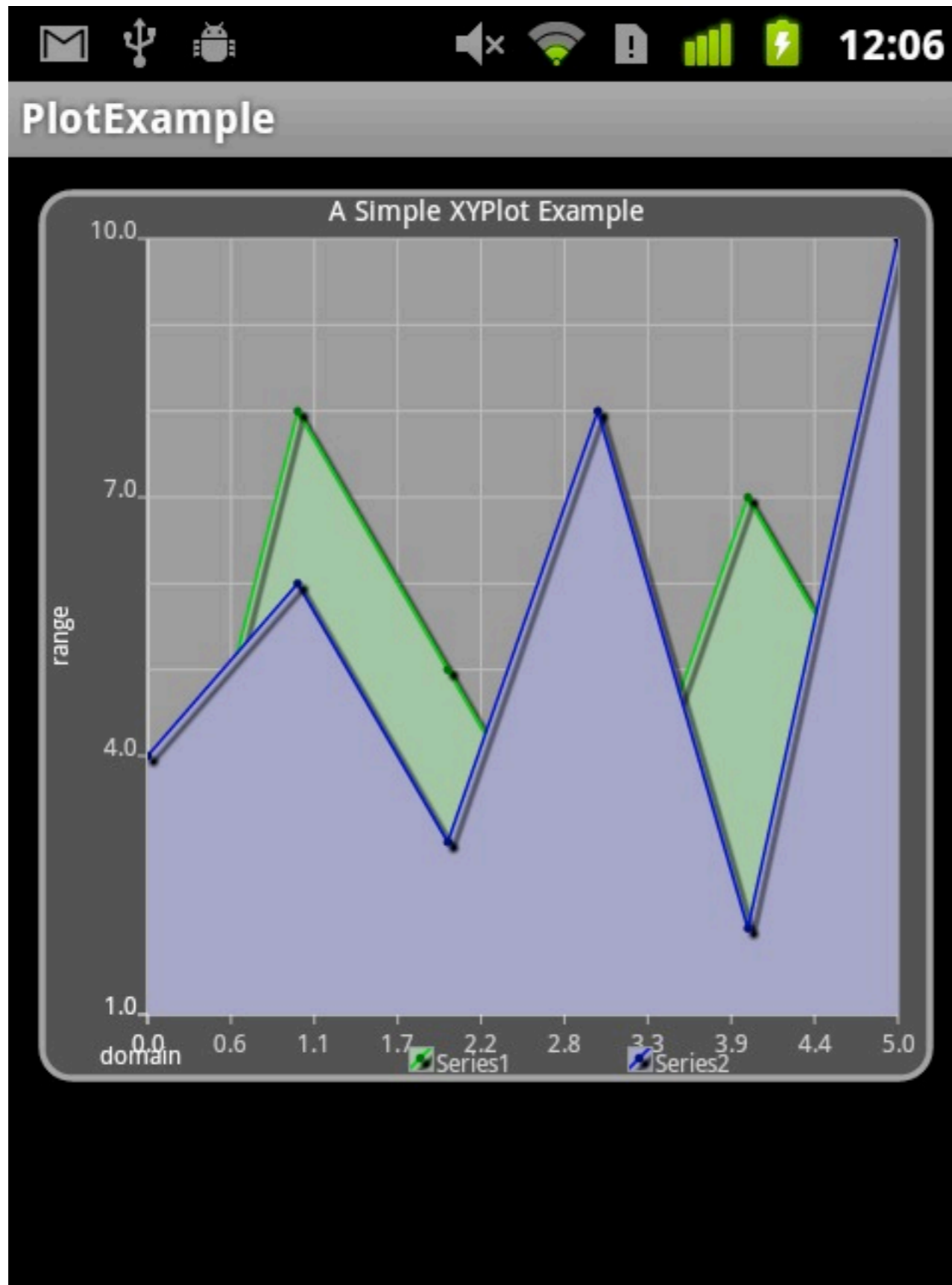
But free

Has complete examples on-line

Getting Started

<http://androidplot.com/wiki/Quickstart>

AndroidPlot Example



Layout

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<com.androidplot.xy.XYPlot
    android:id="@+id/mySimpleXYPlot"
    android:layout_width="fill_parent"
    android:layout_height="300dip"
    android:layout_marginTop="10dip"
    android:layout_marginLeft="10dip"
    android:layout_marginRight="10dip"
    title="A Simple XYPlot Example"/>
</LinearLayout>
```

Plot Example

```
public class PlotExample extends Activity {
    private XYPlot mySimpleXYPlot;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        mySimpleXYPlot = (XYPlot) findViewById(R.id.mySimpleXYPlot);
        Number[] series1Numbers = { 1, 8, 5, 2, 7, 4 };
        Number[] series2Numbers = { 4, 6, 3, 8, 2, 10 };
        XYSeries series1 = new SimpleXYSeries(Arrays.asList(series1Numbers),
            SimpleXYSeries.ArrayFormat.Y_VALS_ONLY, "Series1");
        XYSeries series2 = new SimpleXYSeries(Arrays.asList(series2Numbers),
            SimpleXYSeries.ArrayFormat.Y_VALS_ONLY, "Series2");
    }
}
```

PlotExample Continued

```
LineAndPointFormatter series1Format = new LineAndPointFormatter(  
    Color.rgb(0, 200, 0), // line color  
    Color.rgb(0, 100, 0), // point color  
    Color.rgb(150, 190, 150)); // fill color (optional)  
mySimpleXYPlot.addSeries(series1, series1Format);  
mySimpleXYPlot.addSeries(  
    series2,  
    new LineAndPointFormatter(Color.rgb(0, 0, 200),  
    Color.rgb(0, 0, 100), Color.rgb(150, 150, 190)));  
  
mySimpleXYPlot.setTicksPerRangeLabel(3);  
}  
}
```


Tab View

Resources

Tab tutorial

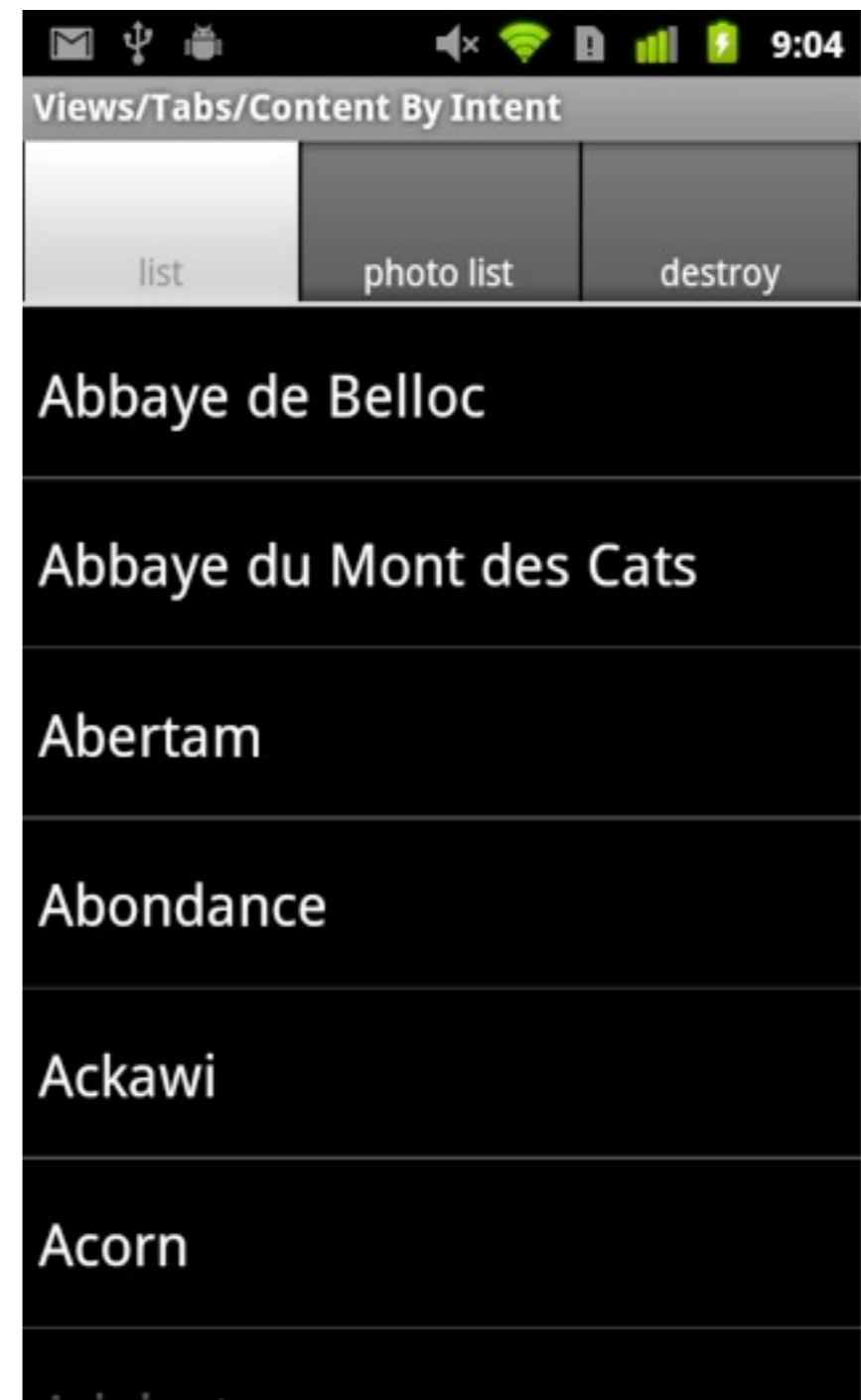
<http://developer.android.com/resources/tutorials/views/hello-tabwidget.html>

API Demo

Tabs1.java

Tabs2.java

Tabs3.java



The Players

TabActivity

Activity you subclass

TabHost

Contains tabs and FrameLayout that displays tab content

Controls tabs & switch between them

TabActivity has one TabHost

TabSpec

Used to specify tab

(Builder pattern)

TabWidget

Widget that displays tabs

Contained by TabHost

Major Parts



Tab has

Indicator

Text displayed on screen

Option icon

Content

Displayed below TabWidget
in FrameLayout

Tag

Used internally by TabHost
Keeps track of tabs



Generating content for a tab

Each tab uses an activity for content

Each tab uses a layout for content

Each tab uses same Factory to generate content

TabActivity Layout

Optional

```
<?xml version="1.0" encoding="utf-8"?>
<TabHost xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@android:id/tabhost"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:padding="5dp">
        <TabWidget
            android:id="@android:id/tabs"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content" />
        <FrameLayout
            android:id="@android:id/tabcontent"
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:padding="5dp" />
    </LinearLayout>
</TabHost>
```

Example - Using Activities for Content



Cascading method calls

```
TabHost.TabSpec rightSpec = tabHost.newTabSpec("albums")  
    .setIndicator("Albums")  
    .setContent(new Intent(this, Right.class));
```



```
TabHost.TabSpec rightSpec = tabHost.newTabSpec("albums");  
rightSpec.setIndicator("Albums");  
rightSpec.setContent(new Intent(this, Right.class));
```

TabActivity

```
public class TabTopLevel extends TabActivity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TabHost tabHost = getTabHost();

        TabHost.TabSpec leftSpec = tabHost
            .newTabSpec("artists")
            .setIndicator("Cats",
                getResources().getDrawable(R.drawable.ic_tab_icons))
            .setContent(new Intent(this, Left.class));
        tabHost.addTab(leftSpec);

        TabHost.TabSpec rightSpec = tabHost.newTabSpec("albums")
            .setIndicator("Albums")
            .setContent(new Intent(this, Right.class));
        tabHost.addTab(rightSpec);
        tabHost.setCurrentTab(1);
    }
}
```

The Content

```
public class Left extends Activity {  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        TextView textview = new TextView(this);  
        textview.setText("This is the Left tab");  
        setContentView(textview);  
    }  
}
```

Details - icons

Need two icons - dark & light

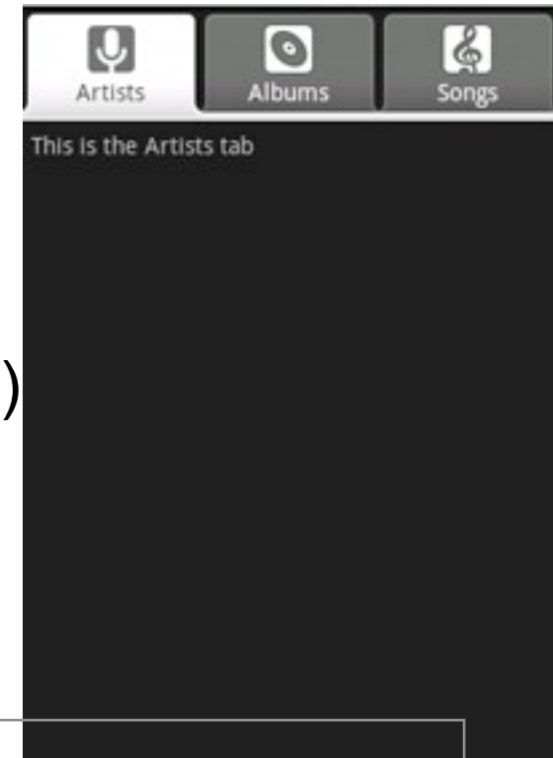
```
getResources().getDrawable(R.drawable.ic_tab_icons))
```

drawable/ic_tab_icons.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
  <item android:drawable="@drawable/ic_tab_artists_grey"
        android:state_selected="true" />
  <item android:drawable="@drawable/ic_tab_artists_white" />
</selector>
```

drawable/ic_tab_artists_grey.png

Actual image file



Details - Manifest file

Each Activity must be listed

```
<activity android:name=".Left"
          android:label="Left">
</activity>
```

Hiding the title bar

```
<activity android:name=".TabTopLevel"
          android:label="@string/app_name"
          android:theme="@android:style/Theme.NoTitleBar">
<intent-filter>
```

Example - Using TabConnectionFactory

```
public class TabTopLevel extends TabActivity implements
    TabHost.TabConnectionFactory {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TabHost tabHost = getTabHost();

        TabHost.TabSpec leftSpec = tabHost
            .newTabSpec("artists")
            .setIndicator("Cats",
                getResources().getDrawable(R.drawable.ic_tab_icons))
            .setContent(this);
        tabHost.addTab(leftSpec);

        TabHost.TabSpec rightSpec = tabHost.newTabSpec("albums")
            .setIndicator("Albums").setContent(this);
        tabHost.addTab(rightSpec);
        tabHost.setCurrentTab(1);
    }
}
```

The Factory method

```
public View createTabContent(String tag) {
    final TextView generatedView = new TextView(this);
    if (tag.equals("artists")) {
        generatedView.setText("You selected Artists");
    } else {
        generatedView.setText("Here are the albums");
    }
    return generatedView;
}
```