

EMBEDDED SYSTEMS PROGRAMMING 2016-17

Application Tip: Switching UIs

THE PROBLEM

- How to switch from one UI to another
- Each UI is associated with a distinct class that controls it
- Solution shown: two UIs, hence two classes. A button in each UI triggers the switch to the other UI

THE TIP

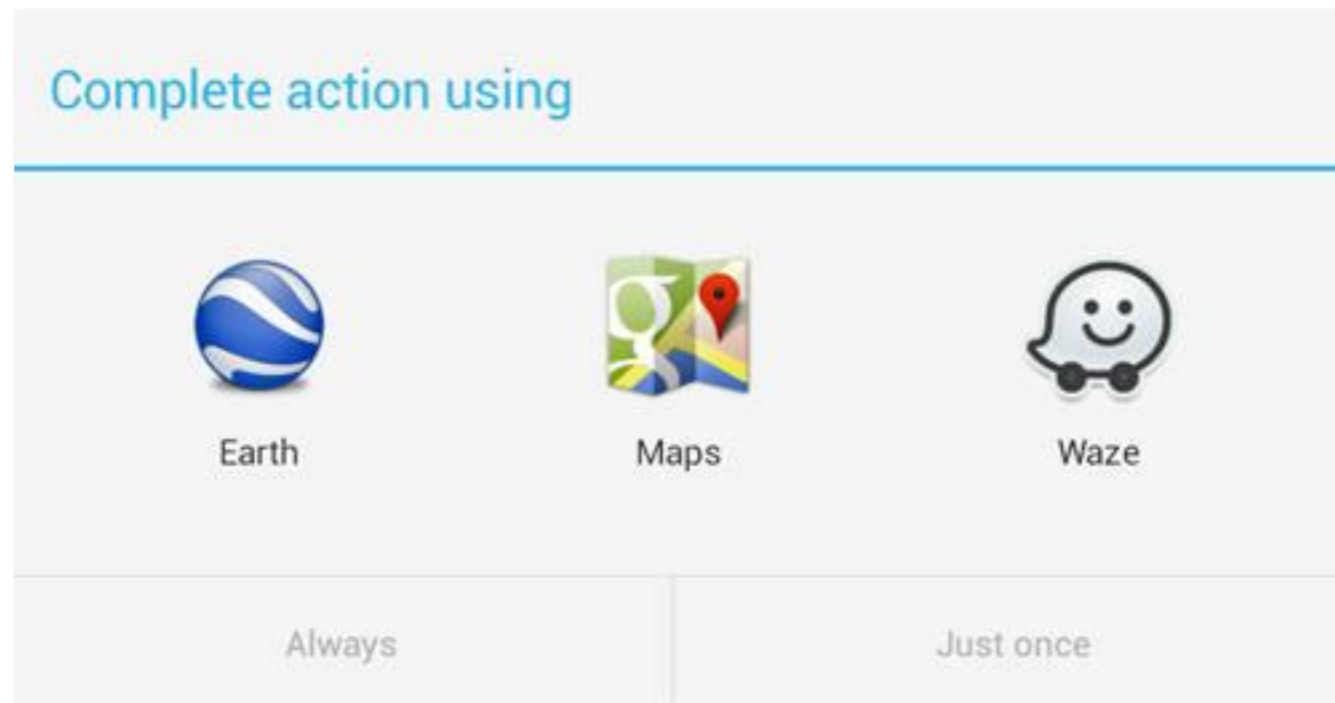
- Two classes derived from `Activity`
- The UIs associated with the classes are defined declaratively, hence there are two XML layout files
- The two classes invoke each other via `intents`

INTENTS (1/3)

- **Intent: abstract request for an action to be performed**
- **Explicit intent: requires an exact component (e.g., a class) to perform the operation**
- **Implicit intent: does not specify a class, but includes enough information for the system to determine which of the available components is to be invoked (intent resolution)**

INTENTS (2/3)

- The intent resolution process may find more than one suitable component: in this case, the final choice is up to the user



Action requested: display a location on the map

INTENTS (3/3)

- Intents have a flexible (and complex) syntax; <http://developer.android.com/guide/components/intents-filters.html> provides full details
- To invoke a specific component (such as an activity), simply build an intent specifying
 - a **context** of the application package implementing the component,
 - the **component class**

CODE (1/6)

- Source files:
 - `Activity1.java`,
 - `Activity2.java`
- Layouts:
 - `activity_1.xml`,
 - `activity_2.xml`

CODE (2/6)

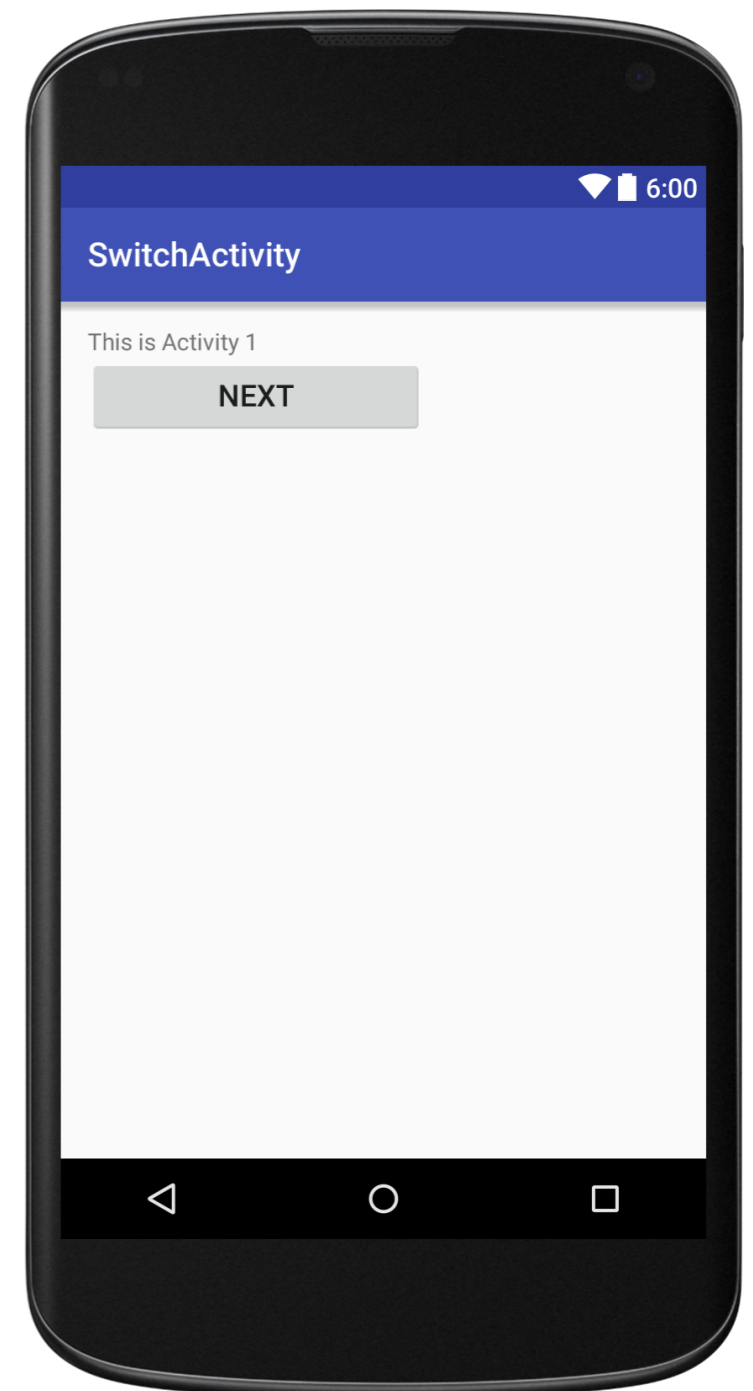
• activity_1.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="it.unipd.dei.esp1516.switchactivity.Activity1">

    <TextView android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="This is Activity 1"
        android:id="@+id/textView" />

    <Button android:text="Next"
        android:id="@+id/Button01"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:layout_below="@+id/textView"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">
</Button>

</RelativeLayout>
```



CODE (3/6)

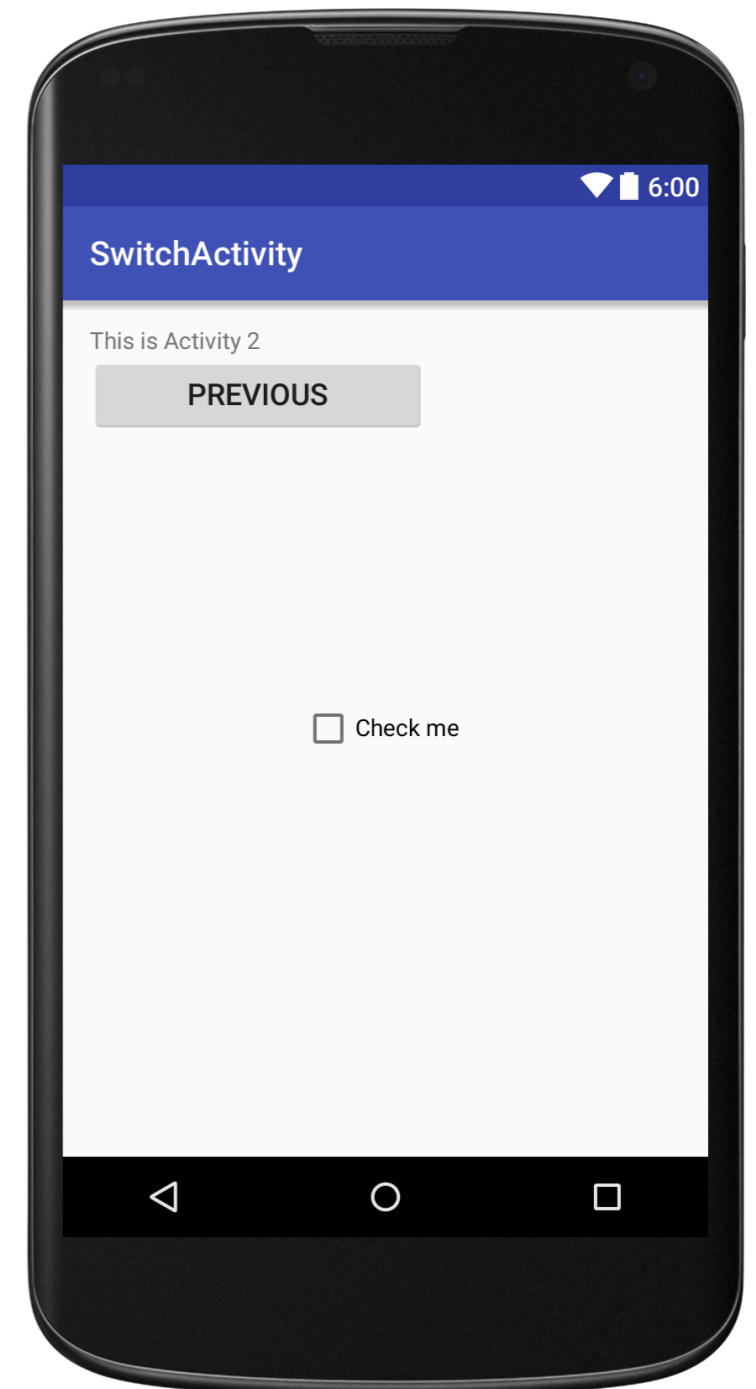
• activity_2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

...
  <TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="This is Activity 2"
    android:id="@+id/textView2" />

  <Button
    android:id="@+id/Button02"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:text="Previous"
    android:textSize="18sp"
    android:layout_below="@+id/textView2"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true">
  </Button>

  <CheckBox
    android:id="@+id/plain_cb"
    android:layout_width="200px"
    android:layout_height="wrap_content"
    android:text="Check me"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true" />
</RelativeLayout>
```



CODE (4/6)

● Activity1.java

```
package it.unipd.dei.esp1516.switchactivity;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class Activity1 extends AppCompatActivity
{
    /** Called when the activity is first created. */
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_1);

        Button next = (Button) findViewById(R.id.Button01);
        next.setOnClickListener(new View.OnClickListener() {
            public void onClick(View view) {
                Intent myIntent = new Intent(view.getContext(), Activity2.class);
                startActivityForResult(myIntent, 0);
            }
        });
    }
}
```


CODE (5/6)

● Activity2.java

```
package it.unipd.dei.esp1516.switchactivity;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class Activity2 extends AppCompatActivity
{
    /** Called when the activity is first created. */
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_2);

        Button next = (Button) findViewById(R.id.Button02);
        next.setOnClickListener(new View.OnClickListener() {
            public void onClick(View view) {
                Intent intent = new Intent();
                setResult(RESULT_OK, intent);
                finish();
            }
        });
    }
}
```

CODE (6/6)

- Both activities should be declared in the `AndroidManifest.xml` file

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="it.unipd.dei.esp1516.switchactivity">

    <application
        android:allowBackup="false"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <activity android:name=".Activity1">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name=".Activity2"/>

    </application>

</manifest>
```


LAST MODIFIED: MARCH 20, 2017

COPYRIGHT HOLDER: CARLO FANTOZZI (FANTOZZI@DEI.UNIPD.IT)
LICENSE: CREATIVE COMMONS ATTRIBUTION SHARE-ALIKE 3.0