Android
Notifications

Victor Matos
Cleveland State University

Notes are based on:
    Android Developers
    http://developer.android.com/index.html

Notifications

What is a Notification?

A *notification* is a short message briefly displayed on the *status line*.

It typically announces the happening of an special event for which a trigger has been set.

After opening the *Notification Panel* the user may choose to click on a selection and execute an associated activity.

Notifications

What is a Notification?

5554: Golfing at Manakiki sat. at 1:00pm

Notification shown on the status line

Drag down

November 21, 2009 8:30 PM

Android Clear

Notifications

5554
2 unread messages. 8:29 PM

Click on *Notification Panel* to execute associated application

555-4 8:31 PM

5554: Remember to buy bread and milk
Sent: 8:27PM

5554: Golfing at Manakiki sat. at 1:00pm
Sent: 8:29PM

Type to compose Send

Notifications

Notification Manager

This class notifies the user of events that happen in the background.

Notifications can take different forms:

1. A persistent icon that goes in the status bar and is accessible through the launcher, (when the user selects it, a designated Intent can be launched),

2. Turning on or flashing LEDs on the device, or

3. Alerting the user by flashing the backlight, playing a sound, or vibrating.

Notifications – Since Jelly Bean 4.0

Base Layout

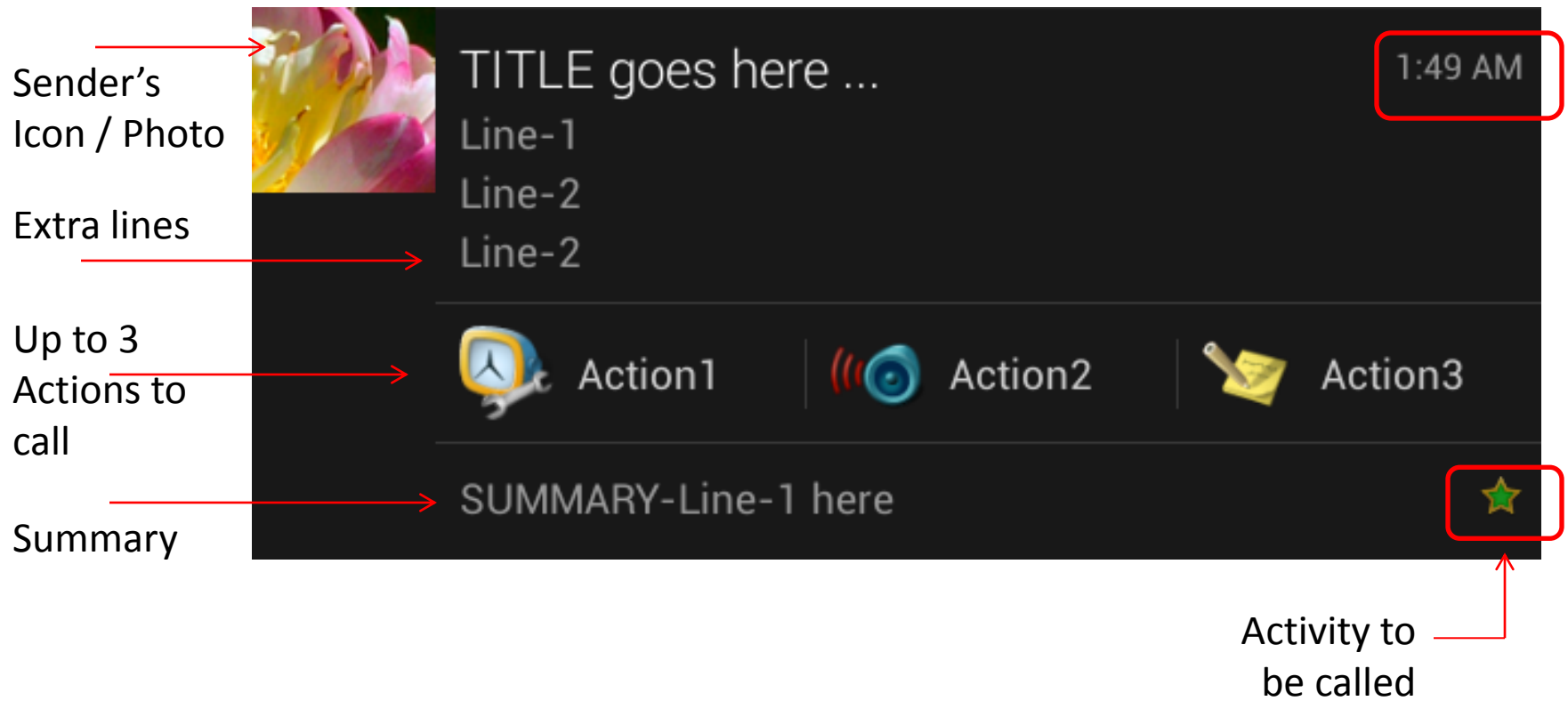All notifications include in their MINIMUM configurations three parts:

1. the sending application's notification icon or the sender's photo a notification title and message
2. a timestamp
3. a secondary icon to identify the sending application when the senders image is shown for the main icon

Optional Entries
1. Additional lines
2. Up to three actions
3. A summary line

Notifications – Since Jelly Bean 4.0

Extended Layout

Sender's
Icon / Photo

Extra lines

Up to 3
Actions to
call

Summary

TITLE goes here ...
Line-1
Line-2
Line-2

1:49 AM

Action1   Action2   Action3

SUMMARY-Line-1 here

Activity to
be called

Notifications

Notification Manager

You do not instantiate this class directly; instead, retrieve it through **getSystemService (String)**.

Example:

```
String servName = Context.NOTIFICATION_SERVICE;

notificationManager = (NotificationManager)
                    getSystemService (servName);
```

Notifications
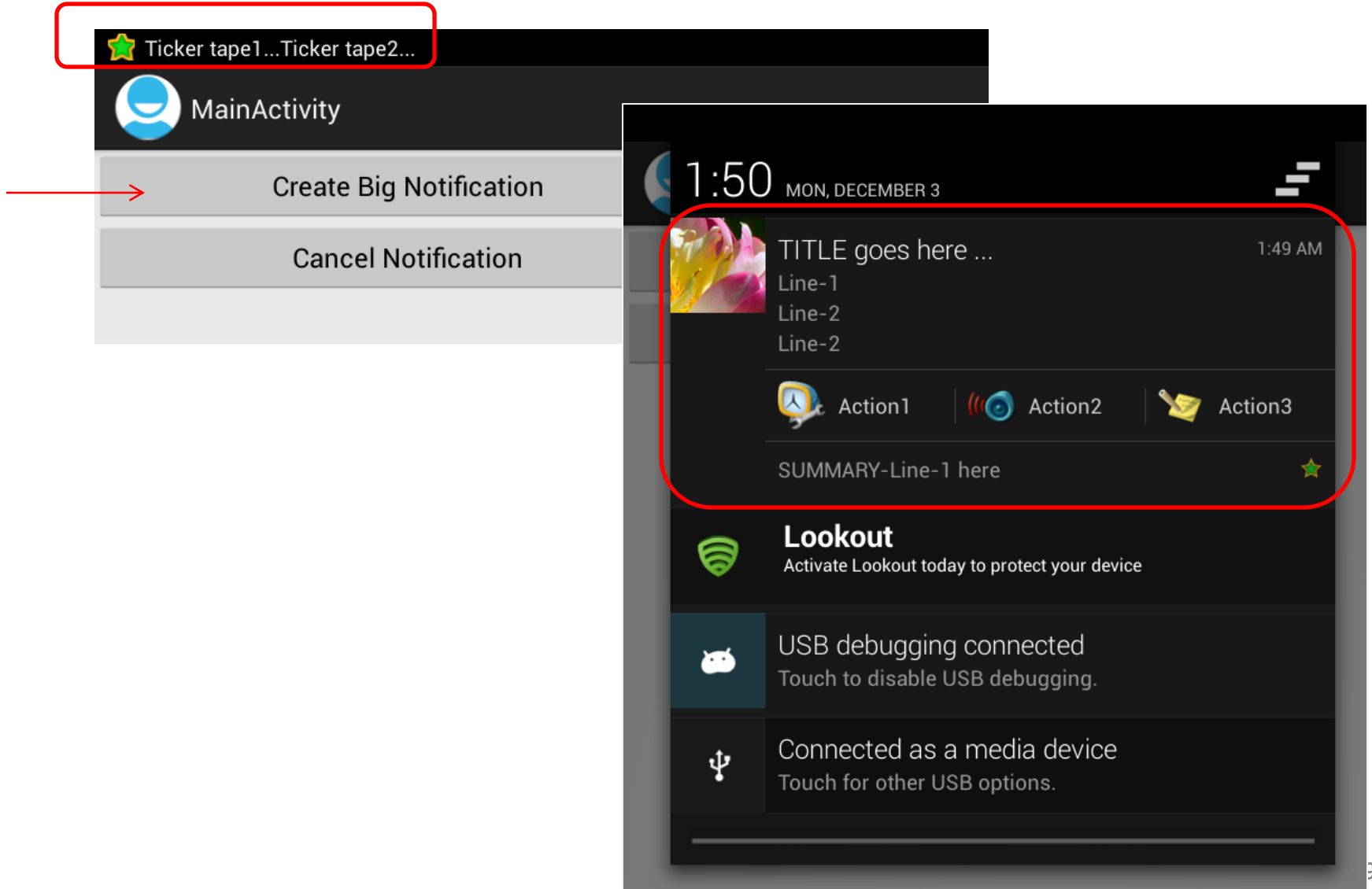
Notification Builder

A convenient way to set up various fields of a **Notification**

Example:

```
Notification noti = new Notification.Builder(Context)
    .setContentTitle("Important message for you...")
    .setContentText(subject)
    .setSmallIcon(R.drawable.new_mail)
    .setLargeIcon(aBitmap)
    .build();
```
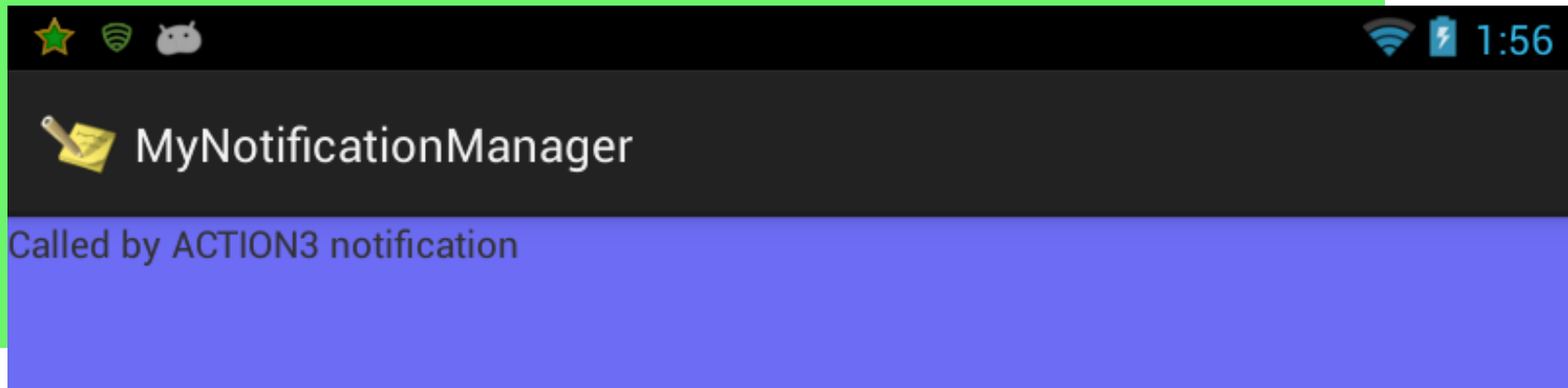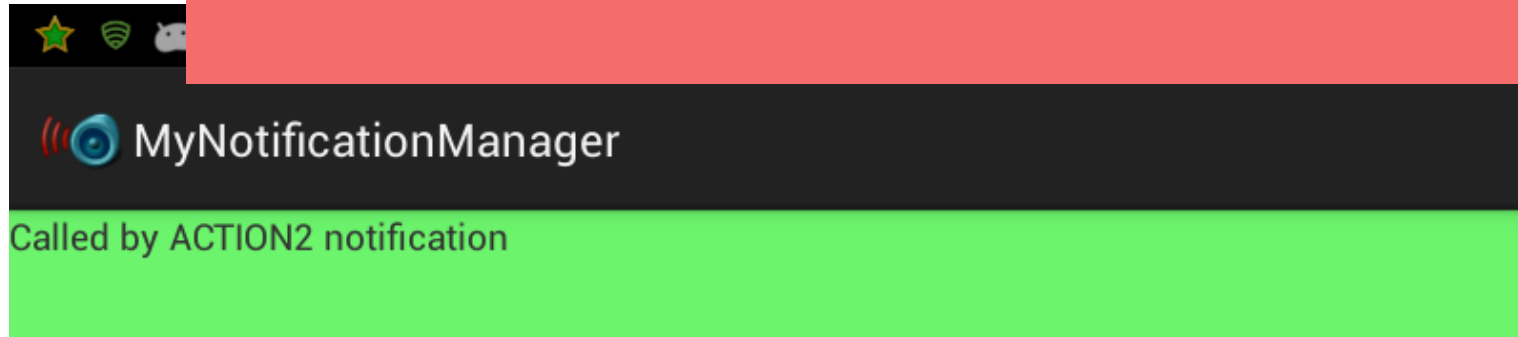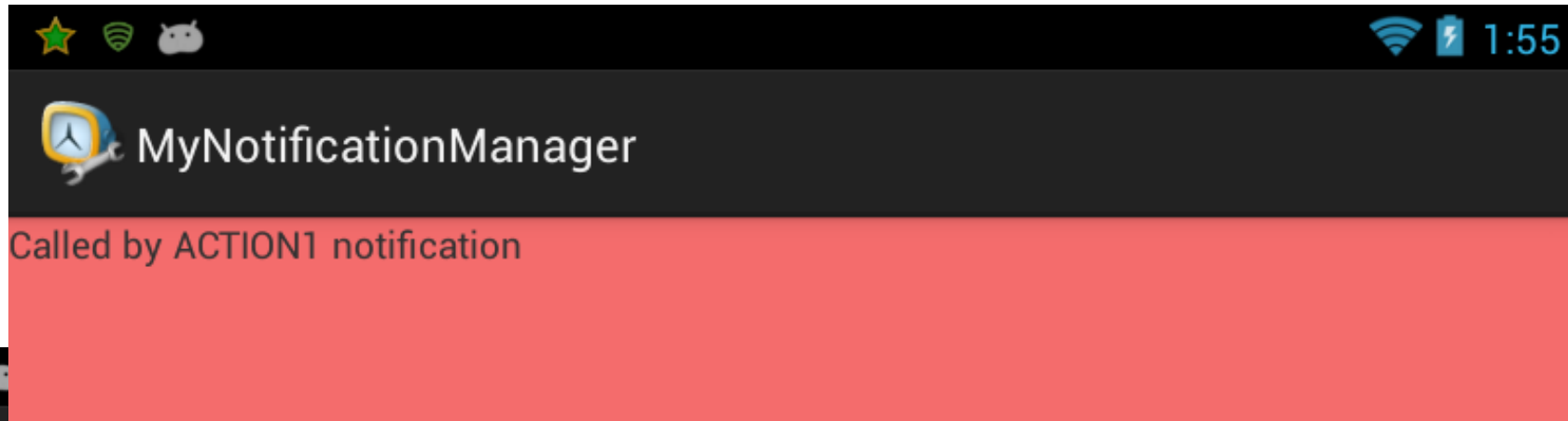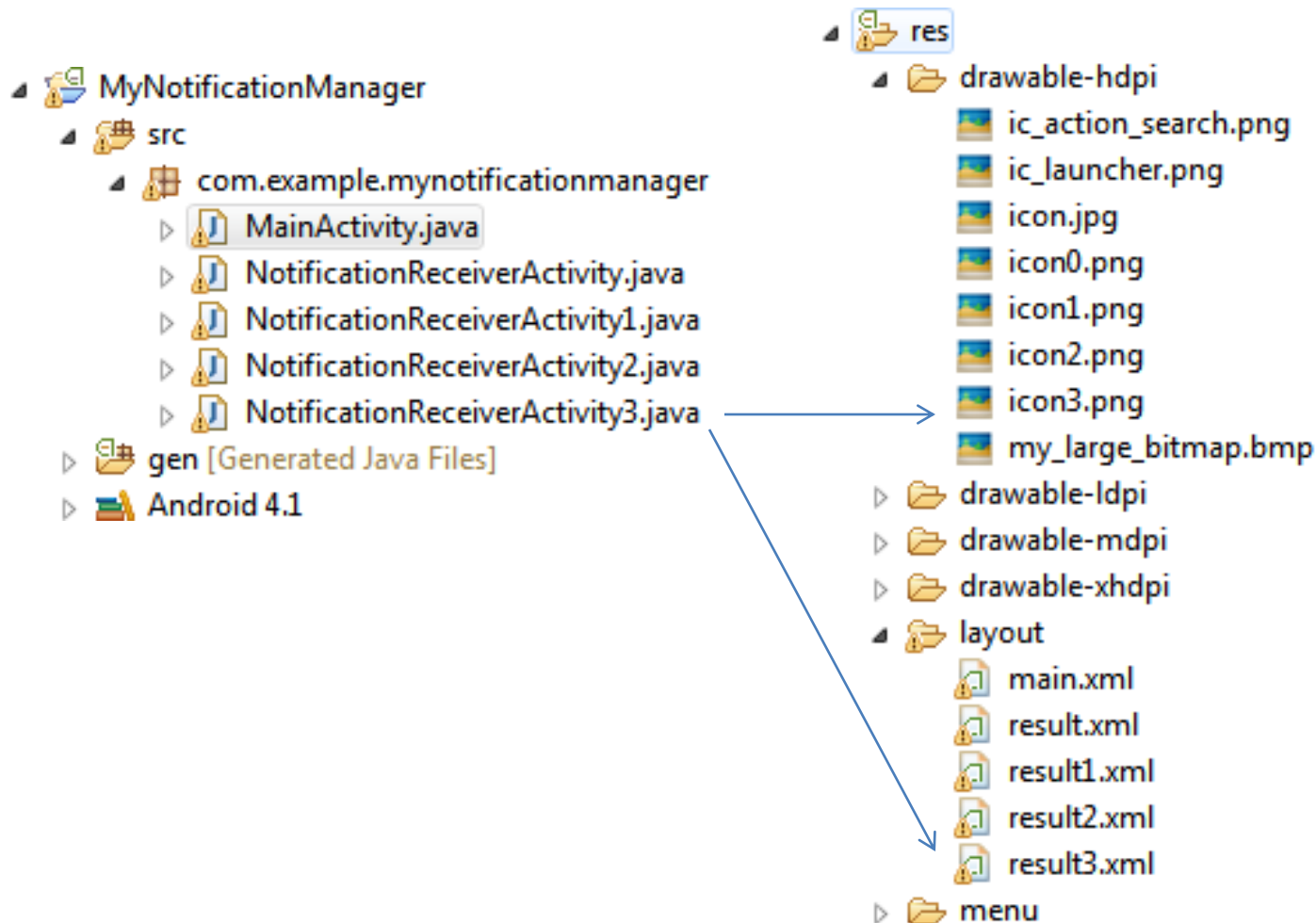
Notifications

Example

Notifications

Example

Notifications

Example

Notifications

Example: MainActivity

```java
package com.example.mynotificationmanager;
import . . .

public class MainActivity extends Activity implements OnClickListener {
  NotificationManager notificationManager;
  final int NOTIFICATION_ID = 12345;

  @Override
  public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    setContentView(R.layout.main);
    findViewById(R.id.btnBig).setOnClickListener(this);
    findViewById(R.id.btnCancel).setOnClickListener(this);

  }// onCreate

  @SuppressLint("NewApi")
  public void createBigNotification(View view) {

    Intent intent = new Intent(this, NotificationReceiverActivity.class);
    intent.putExtra("callerIntent", "main");
    intent.putExtra("notificationID", NOTIFICATION_ID);
    PendingIntent pIntent = PendingIntent.getActivity(this, 0, intent, 0);
```

Notifications

Example: MainActivity

```java
// better way to do previous work
PendingIntent pIntent1 = makePendingIntent(
    NotificationReceiverActivity1.class, "Action1");
PendingIntent pIntent2 = makePendingIntent(
    NotificationReceiverActivity2.class, "Action2");
PendingIntent pIntent3 = makePendingIntent(
    NotificationReceiverActivity3.class, "Action3");

// a bitmap to be added in the notification view
Bitmap myBitmap = BitmapFactory.decodeResource(getResources(),
    R.drawable.my_large_bitmap);

Notification.Builder baseNotification = new Notification.Builder(this)
    .setContentTitle("TITLE goes here ...")
    .setContentText("Second Line of text goes here")
    .setTicker("Ticker tape1...Ticker tape2...")
    .addAction(R.drawable.icon1, "Action1", pIntent1)
    .addAction(R.drawable.icon2, "Action2", pIntent2)
    .addAction(R.drawable.icon3, "Action3", pIntent3)
    .setSmallIcon(R.drawable.icon0)
    .setLargeIcon(myBitmap)
    .setLights(0xffcc00, 1000, 500)
    .setContentIntent(pIntent) ;
```

Notifications

Example: MainActivity

```java
    Notification noti = new Notification.InboxStyle(baseNotification)
                .addLine("Line-1")
                .addLine("Line-2")
                .addLine("Line-2")
                .setSummaryText("SUMMARY-Line-1 here")
                .build();

    notificationManager = (NotificationManager)
                        getSystemService(NOTIFICATION_SERVICE);
    // Hide the notification after its selected
    noti.flags |= Notification.FLAG_AUTO_CANCEL;
    // notification ID is 12345
    notificationManager.notify(12345, noti);

}// createBigNotification

@Override
public void onClick(View v) {
  switch (v.getId()) {
  case R.id.btnBig:
    createBigNotification(v);
    break;
```

Notifications

Example: MainActivity

```java
      case R.id.btnCancel :
        try {
          if ( notificationManager != null){
            notificationManager.cancel(NOTIFICATION_ID);
          }
        } catch (Exception e) {
          Log.e("<<MAIN>>", e.getMessage() );
        }
        break;

      }
    }// onClick

  public PendingIntent makePendingIntent(Class partnerClass, String callerName)
{
    Intent intent = new Intent(this, partnerClass);
    intent.putExtra("callerIntent", callerName);
    intent.putExtra("notificationID", NOTIFICATION_ID);
    PendingIntent pIntent = PendingIntent.getActivity(this, 0, intent, 0);
    return pIntent;
  }

}// class
```

Notifications

Example: NotificationReceiverActivity

```java
package com.example.mynotificationmanager;

import . . .

public class NotificationReceiverActivity extends Activity {
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.result);

    String callerName = getIntent()
                      .getStringExtra("callerIntent");

    Toast.makeText(this, "Called by: " + callerName, 1).show();
  }
}
```

Notifications

Example: Manifest

```xml
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.mynotificationmanager"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="14" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" ><activity
            android:name=".MainActivity"
            android:label="@string/title_activity_main" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

        <activity android:name=".NotificationReceiverActivity"
                android:icon="@drawable/icon0"  >
        </activity>
```

Notifications

Example: Manifest

```
  <activity android:name=".NotificationReceiverActivity1"
          android:icon="@drawable/icon1" >
  </activity>

  <activity android:name=".NotificationReceiverActivity2"
         android:icon="@drawable/icon2" >
  </activity>

  <activity android:name=".NotificationReceiverActivity3"
         android:icon="@drawable/icon3" >
  </activity>

 </application>

</manifest>
```

Notifications

Example - **Layout**: main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >

    <Button
        android:id="@+id/btnBig"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Create Big Notification"
        android:ems="20" >
    </Button>

    <Button
        android:id="@+id/btnCancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Cancel Notification"
        android:ems="20" >
    </Button>

</LinearLayout>
```

Notifications

Example - **Layout**: main.xml

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

    <TextView
        android:id="@+id/txtMsg"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Called from the MASTER notification" >
    </TextView>

</LinearLayout>
```

Notifications (Before SDK 4.0)

Notification

This class represents how a persistent notification is to be presented to the user using the NotificationManager.

public Notification **(int** icon**, CharSequence** tickerText**, long** when**)**

Parameters

icon　　　　The resource id of the icon to put in the status bar.

tickerText　The text that flows by in the status bar when the notification first activates.

when　　　The time to show in the time field.
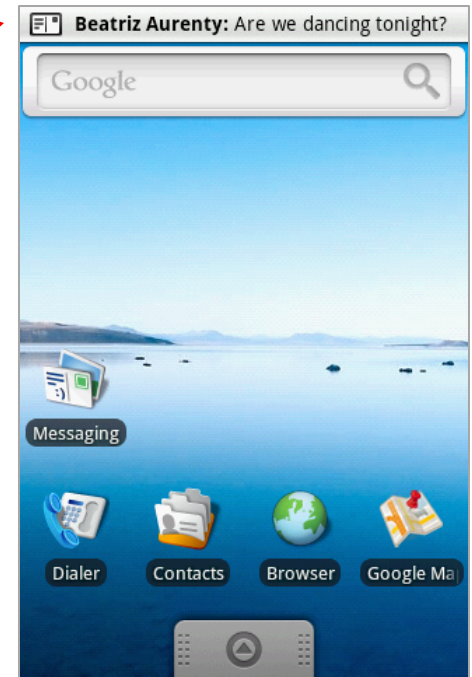　　　　　　In the System.currentTimeMillis timebase.

Notifications (Before SDK 4.0)

Notification - Methods

public void notify **(int** id**, Notification** notification**)**

Places a persistent notification on the status bar.

Parameters

id   An identifier for this notification unique within your application.

notification A Notification object describing how to notify the user, other than the view you're providing.
Must not be null.
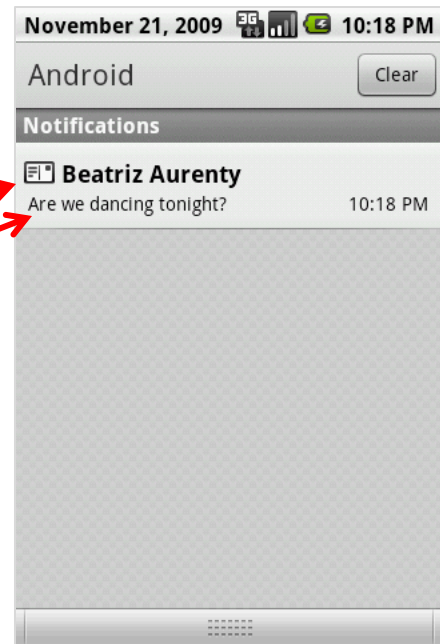
Notifications (Before SDK 4.0)

Notification – Methods

---

public void setLatestEventInfo **(**
        Context context**, CharSequence** contentTitle**,**
        CharSequence contentText**, PendingIntent** contentIntent**)**

Sets the contentView field to be a view with the standard "Latest Event" layout.

Parameters

context	The context for your application / activity.
contentTitle	The title that goes in the expanded entry.
contentText	The text that goes in the expanded entry.
contentIntent	The intent to launch when the user clicks the expanded notification.

Notifications (Before SDK 4.0)

Notification – Methods

public void cancel **(int** id **)**
public void cancelAll ( )

Cancel a previously shown notification. If it's transient, the view will be hidden. If it's persistent, it will be removed from the status bar.
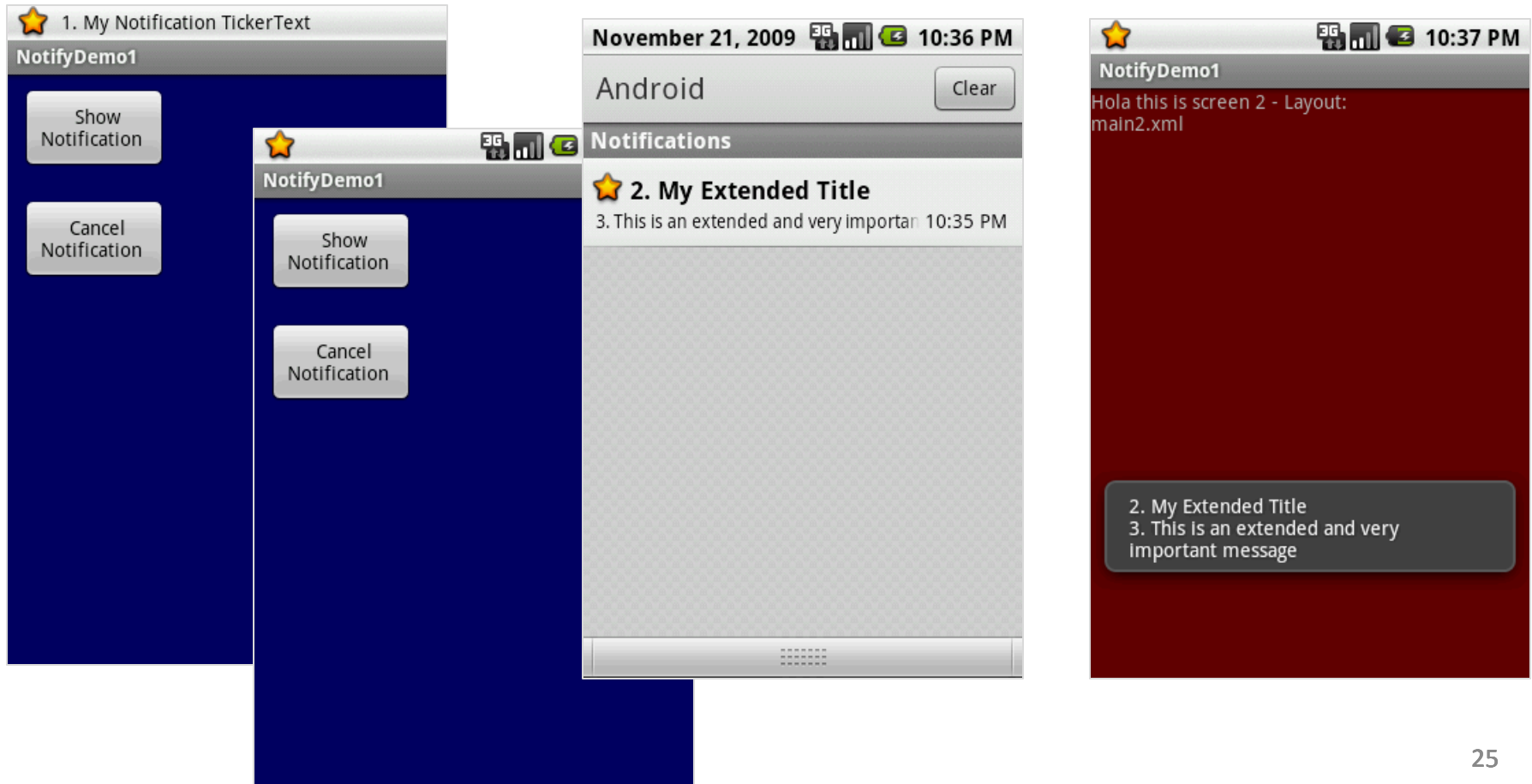
Parameters
Id   An identifier for this notification unique within your application.

Notifications (Before SDK 4.0)

Example.

Produce a notification. Allow the user to click on the Notification Panel and execute appropriate activity to attend the message.

Notifications (Before SDK 4.0)

Example - Layouts

main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
 android:id="@+id/myLinearLayout1"
 android:layout_width="fill_parent"
 android:layout_height="fill_parent"
 android:background="#ff000066"
 android:orientation="vertical"
 xmlns:android="http://schemas.android.com/apk/res/android" >

<Button
 android:id="@+id/btnGo"
 android:layout_width="106dp"
 android:layout_height="61dp"
 android:layout_margin="10dp"
 android:text=" Show Notification " >
</Button>
<Button
 android:id="@+id/btnStop"
 android:layout_width="106dp"
 android:layout_height="61dp"
 android:layout_margin="10dp"
 android:text=" Cancel Notification " >
</Button>

</LinearLayout>
```

main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
 android:id="@+id/main2LinLayout"
 android:layout_width="fill_parent"
 android:layout_height="fill_parent"
 android:background="#ff660000"
 android:orientation="vertical"
 xmlns:android="http://schemas.android.com/apk/res/android"
>

<TextView
 android:id="@+id/widget29"
 android:layout_width="251dp"
 android:layout_height="69dp"
 android:text="Hola this is screen 2 - Layout:main2.xml"
>
</TextView>

</LinearLayout>
```

Notifications (Before SDK 4.0)

Example – Manifest

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="cis493.demos"
    android:versionCode="1"
    android:versionName="1.0">

    <application android:icon="@drawable/icon"
                android:label="@string/app_name">

    <activity android:name=".NotifyDemo1"
            android:label="@string/app_name">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>

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

  </application>

  <uses-sdk android:minSdkVersion="4" />

</manifest>
```

/drawable

btn_star_big_on_selected.png

Note:
Obtain the icon from the folder

C:\Android\platforms\android-1.x\data\res\drawable

Notifications (Before SDK 4.0)

Example – Create & Cancel a Notification

```java
package cis493.demos;

import . . .

public class NotifyDemo1 extends Activity {
    Button btnGo;
    Button btnStop;
    int  notificationId = 1;
    NotificationManager notificationManager;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
```

Notifications (Before SDK 4.0)

Example – Create & Cancel a Notification

```java
btnGo = (Button)findViewById(R.id.btnGo);
btnGo.setOnClickListener(new OnClickListener() {

public void onClick(View v) {
   //define a notification manager
   String serName = Context.NOTIFICATION_SERVICE;
   notificationManager = (NotificationManager)getSystemService(serName);

   //define notification using: icon, text, and timing.
   int icon = R.drawable.btn_star_big_on_selected;
   String tickerText = "1. My Notification TickerText";
   long when = System.currentTimeMillis();
   Notification notification = new Notification(icon, tickerText, when);

   //configure appearance of the notification
   String extendedTitle = "2. My Extended Title";
   String extendedText  = "3. This is an extended and very important message";

   Intent intent = new Intent(getApplicationContext(), NotifyHelper.class);
   intent.putExtra("extendedText", extendedText);
   intent.putExtra("extendedTitle", extendedTitle);
   PendingIntent launchIntent =
       PendingIntent.getActivity(getApplicationContext(),0,intent,0);
```

Notifications (Before SDK 4.0)

Example – Create & Cancel a Notification

```java
        notification.setLatestEventInfo(getApplicationContext(),
                                        extendedTitle, extendedText,
                                        launchIntent);
        //trigger notification
        notificationId = 1;
        notificationManager.notify(notificationId, notification);

     }//click
    });

    btnStop = (Button)findViewById(R.id.btnStop);
    btnStop.setOnClickListener(new OnClickListener() {
     public void onClick(View v) {
       //canceling a notification
       notificationId = 1;
       notificationManager.cancel(notificationId);
      }
     });
  }//onCreate
}//NotifyDemo1
```

Notifications (Before SDK 4.0)

Example - SubActivity – Attending the Notification

```
package cis493.demos;

Import. . .

public class NotifyHelper extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
        setContentView(R.layout.main2);

        Intent myData = getIntent();
        // extract the extra-data in the Notification
        String msg = myData.getStringExtra("extendedTitle") + "\n"
                    + myData.getStringExtra("extendedText");

        Toast.makeText(getApplicationContext(), msg, 1).show();
    }
}
```

Notifications

Questions