SetView(getLayoutInflater().inflate
(R.Layout.custom)); XML

Custom Layout

User Interaction

Transitory Message Pop up

Toasts

Toast!

SetGravity(Gravity...)

SetDuration()

Toast.makeText
.show()

Toast

Toast

User Feedback

User Notification

Top device status bar alerting user

Dialogs

Attach Intent

Can

Notification Drawer

View

action

Intent

Notification Area

Small icon | Title | Detail
| Ticker text

notification manager

ID

Notification Builder

NotificationManager Nm = (Nt.mgr) getSystemService(
Context. NOTIFICATION-SERVICE));

nm.notify(ID, notificationBuilder. build());

SetTicker()

SetAutoCancel(true)

SetSmallIcon(R.Drawable...)

SetSound()

SetContentIntent()

SetContent()

SetVibrate()
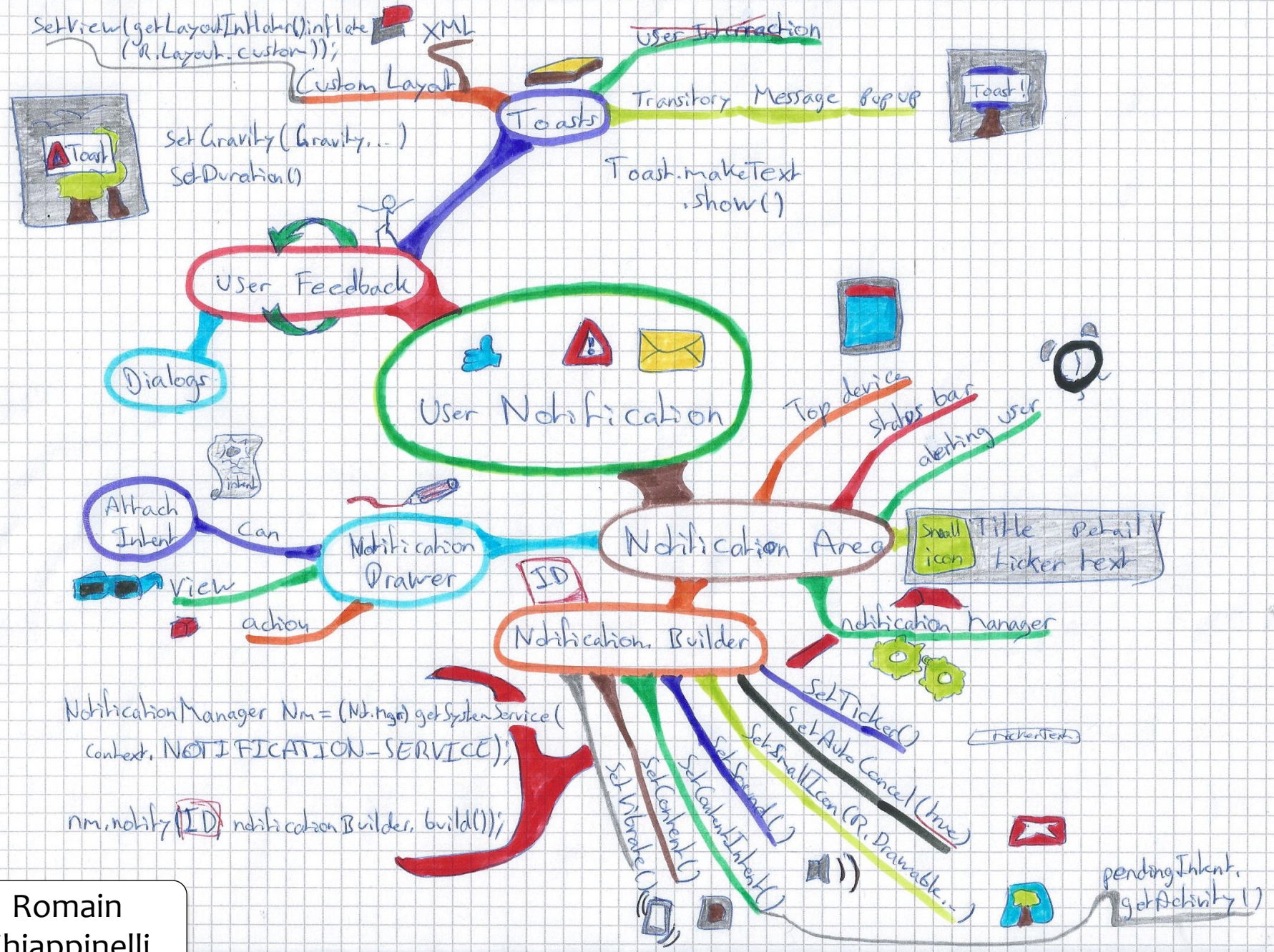
TickerText

pendingIntent.
getActivity()

Romain
Chiappinelli

Programming Handheld Systems

Adam Porter

User Notification

Today's Topics

Toast

Notification Area Notifications

User Notifications

Messages provided to the user outside of the normal UI

User Notifications

These include messages aimed at

User feedback

- Toasts

- Dialogs

Event notification

- Notification Area notifications

Toast

Transitory messages that pop up on the current window

- E.G., to inform user that an operation has completed successfully

Automatically fade into & out of view

No user interaction or response

Creating Toast Notifications

Instantiate a Toast object by calling

Toast.makeText(context, text, duration)
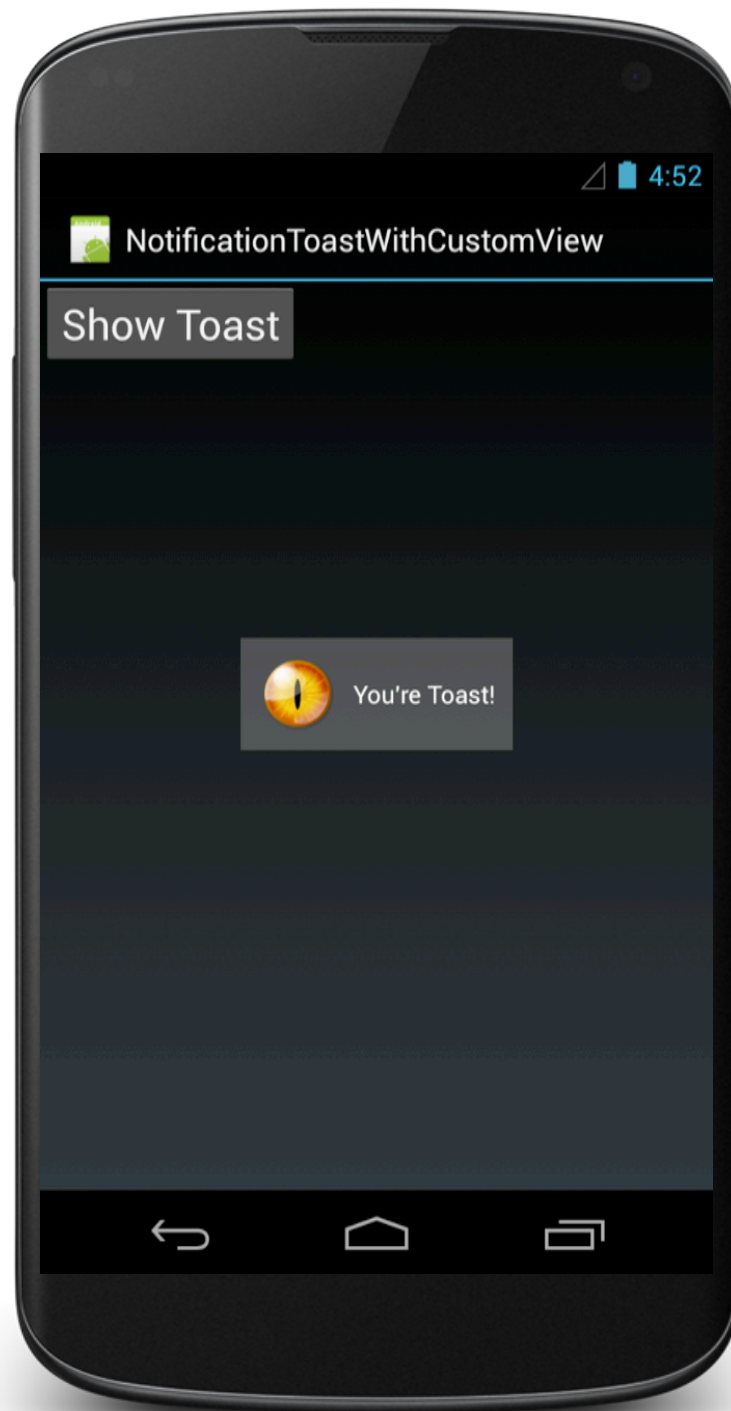
Show toast by calling

Toast.show()
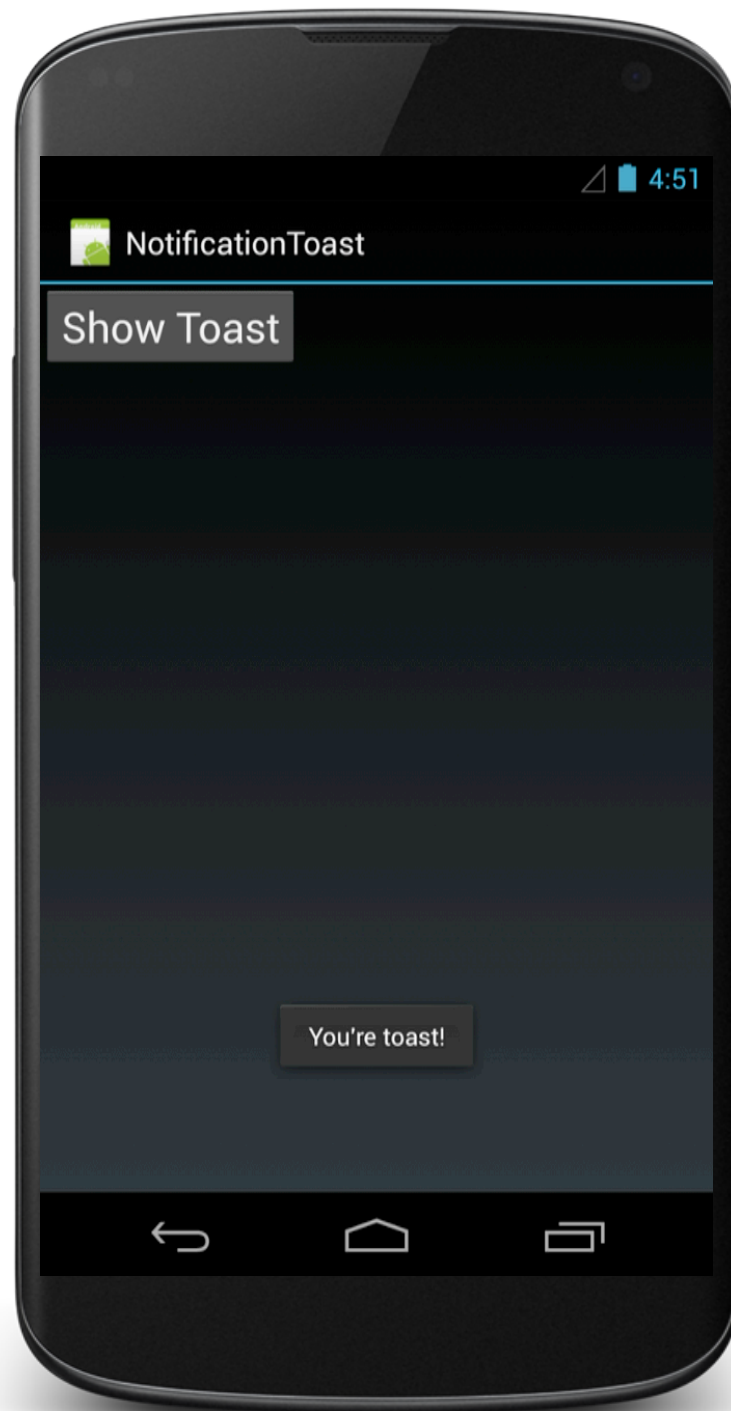
NotificationToast

```java
@Override
public void onClick(View v) {
    Toast.makeText(getApplicationContext(), "You're toast!", Toast.LENGTH_LONG).show();
}
```

Toast with Custom Views

Can create custom layouts in XML

Set them using Toast.setView()

NotificationToastWithCustomView

```java
@Override
public void onClick(View v) {

    Toast toast = new Toast(getApplicationContext());

    toast.setGravity(Gravity.CENTER_VERTICAL, 0, 0);
    toast.setDuration(Toast.LENGTH_LONG);

    toast.setView(getLayoutInflater().inflate(R.layout.custom_toast,null));

    toast.show();
}
```
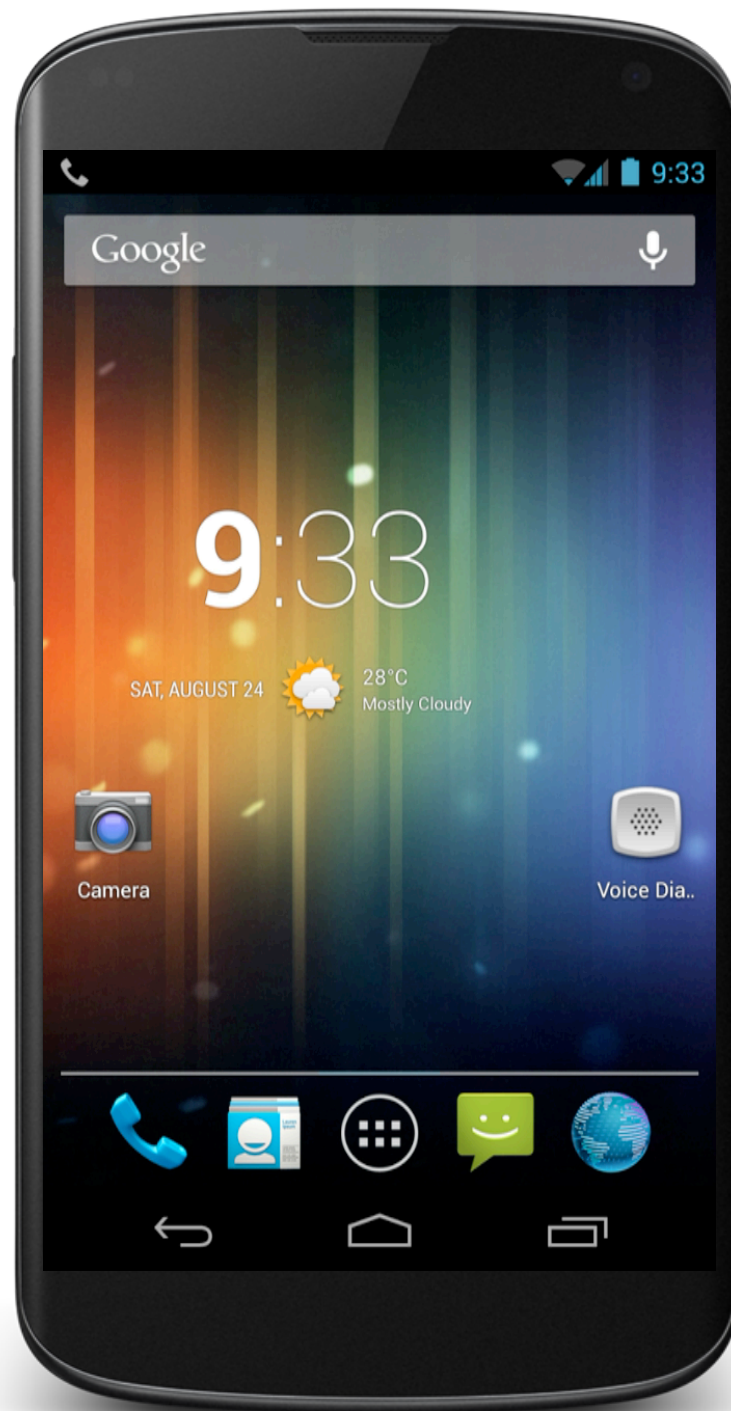
Notification Area Notifications

Android provides the notification area for alerting users about events

It also provides a notification drawer that users can pull down to see more detailed information about notifications

Notification Architecture

Notification

Title, detail, small icon

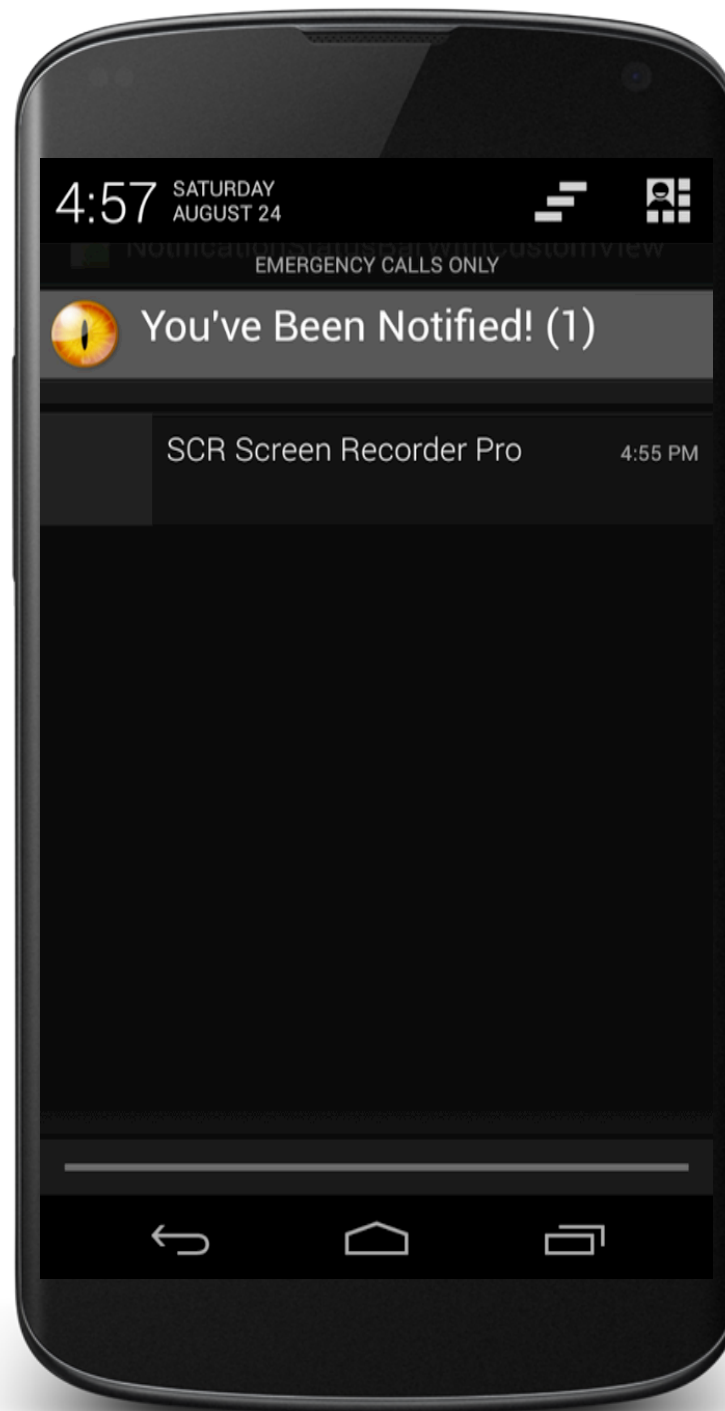## Notification Area

Ticker text, small icon

Notification Drawer

View

Action

Notification Manager

==System service that manages Notifications==

send & cancel notification

⚠ This is a Really, Really, Super Long Notification

NotificationStatusBar

Notify

NotificationStatusBarWithCustomView

EMERGENCY CALLS ONLY

You've Been Notified! (1)

SCR Screen Recorder Pro   4:55 PM

Custom
View

NotificationStatusBarWithCustomView

```java
mNotificationIntent = new Intent(getApplicationContext(),
        NotificationSubActivity.class);
mContentIntent = PendingIntent.getActivity(getApplicationContext(), 0,
        mNotificationIntent, Intent.FLAG_ACTIVITY_NEW_TASK);

final Button button = (Button) findViewById(R.id.button1);
button.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {

        // Define the Notification's expanded message and Intent:

        mContentView.setTextViewText(R.id.text, contentText + " ("
                + ++mNotificationCount + ")");
```

NotificationStatusBarWithCustomView

```java
// Build the Notification

Notification.Builder notificationBuilder = new Notification.Builder(
        getApplicationContext())
    .setTicker(tickerText)
    .setSmallIcon(android.R.drawable.stat_sys_warning)
    .setAutoCancel(true)
    .setContentIntent(mContentIntent)
    .setSound(soundURI)
    .setVibrate(mVibratePattern)
    .setContent(mContentView);

// Pass the Notification to the NotificationManager:
NotificationManager mNotificationManager = (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);
mNotificationManager.notify(MY_NOTIFICATION_ID,
        notificationBuilder.build());

    }
});
```

NotificationToastWithCustomView

```java
@Override
public void onClick(View v) {

    Toast toast = new Toast(getApplicationContext());

    toast.setGravity(Gravity.CENTER_VERTICAL, 0, 0);
    toast.setDuration(Toast.LENGTH_LONG);

    toast.setView(getLayoutInflater().inflate(R.layout.custom_toast,null));

    toast.show();
}
```

Next Time

The BroadcastReceiver Class