Using Big View Styles

THIS LESSON TEACHES YOU TO

1. Set Up the Notification to Launch a New Activity
2. Construct the Big View

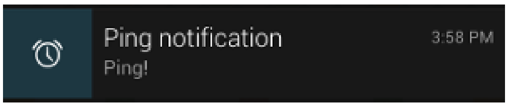
YOU SHOULD ALSO READ

- Notifications API Guide
- Intents and Intent Filters
- Notifications Design Guide

Notifications in the notification drawer appear in two main visual styles, normal view and big view. The big view of a notification only appears when the notification is expanded. This happens when the notification is at the top of the drawer, or the user clicks the notification.

Big views were introduced in Android 4.1, and they're not supported on older devices. This lesson describes how to incorporate big view notifications into your app while still providing full functionality via the normal view. See the Notifications API guide (/guide/topics/ui/notifiers/notifications.html#BigNotify) for more discussion of big views.

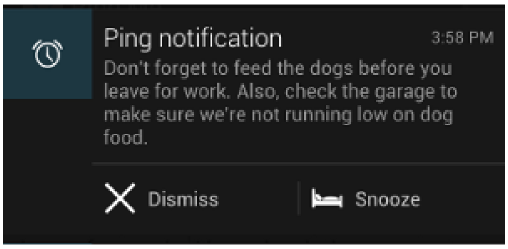
Here is an example of a normal view:



Ping notification 3:58 PM
Ping!

Figure 1. Normal view notification.

Here is an example of a big view:



Ping notification 3:58 PM
Don't forget to feed the dogs before you leave for work. Also, check the garage to make sure we're not running low on dog food.

Figure 2. Big view notification.

In the sample application shown in this lesson, both the normal view and the big view give users access to same functionality:

- The ability to snooze or dismiss the notification.
- A way to view the reminder text the user set as part of the timer.

The normal view provides these features through a new activity that launches when the user clicks the notification. Keep this in mind as you design your notifications—first provide the functionality in the normal view, since this is how many users will interact with the notification.

Set Up the Notification to Launch a New Activity

The sample application uses an `IntentService` `(/reference/android/app/IntentService.html)` subclass (`PingService`) to construct and issue the notification.

In this snippet, the `IntentService` `(/reference/android/app/IntentService.html)` method `onHandleIntent()` `(/reference/android/app/IntentService.html#onHandleIntent(android.content.Intent))` specifies the new activity that will be launched if the user clicks the notification itself. The method `setContentIntent()` `(/reference/android/support/v4/app/NotificationCompat.Builder.html#setContentIntent(android.app.PendingIntent))` defines a pending intent that should be fired when the user clicks the notification, thereby launching the activity.

```
Intent resultIntent = new Intent(this, ResultActivity.class);
resultIntent.putExtra(CommonConstants.EXTRA_MESSAGE, msg);
resultIntent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
        Intent.FLAG_ACTIVITY_CLEAR_TASK);

// Because clicking the notification launches a new ("special") activity,
// there's no need to create an artificial back stack.
PendingIntent resultPendingIntent =
         PendingIntent.getActivity(
         this,
         0,
         resultIntent,
         PendingIntent.FLAG_UPDATE_CURRENT
);

// This sets the pending intent that should be fired when the user clicks the
// notification. Clicking the notification launches a new activity.
builder.setContentIntent(resultPendingIntent);
```

Construct the Big View

This snippet shows how to set up the buttons that will appear in the big view:

```
// Sets up the Snooze and Dismiss action buttons that will appear in the
// big view of the notification.
Intent dismissIntent = new Intent(this, PingService.class);
dismissIntent.setAction(CommonConstants.ACTION_DISMISS);
PendingIntent piDismiss = PendingIntent.getService(this, 0, dismissIntent, 0);

Intent snoozeIntent = new Intent(this, PingService.class);
snoozeIntent.setAction(CommonConstants.ACTION_SNOOZE);
PendingIntent piSnooze = PendingIntent.getService(this, 0, snoozeIntent, 0);
```

This snippet shows how to construct the Builder (/reference/android/support/v4/app/NotificationCompat.Builder.html) object. It sets the style for the big view to be "big text," and sets its content to be the reminder message. It uses addAction() (/reference/android/support/v4/app/NotificationCompat.Builder.html#addAction(int, java.lang.CharSequence, android.app.PendingIntent)) to add the **Snooze** and **Dismiss** buttons (and their associated pending intents) that will appear in the notification's big view:

```
// Constructs the Builder object.
NotificationCompat.Builder builder =
        new NotificationCompat.Builder(this)
        .setSmallIcon(R.drawable.ic_stat_notification)
        .setContentTitle(getString(R.string.notification))
        .setContentText(getString(R.string.ping))
        .setDefaults(Notification.DEFAULT_ALL) // requires VIBRATE permission
        /*
         * Sets the big view "big text" style and supplies the
         * text (the user's reminder message) that will be displayed
         * in the detail area of the expanded notification.
         * These calls are ignored by the support library for
         * pre-4.1 devices.
         */
        .setStyle(new NotificationCompat.BigTextStyle()
                .bigText(msg))
        .addAction (R.drawable.ic_stat_dismiss,
                getString(R.string.dismiss), piDismiss)
        .addAction (R.drawable.ic_stat_snooze,
                getString(R.string.snooze), piSnooze);
```