

Using Proximity Alerts

It's often useful to have your applications react when a user moves toward, or away from, a specific location. Proximity alerts let your applications set triggers that are fired when a user moves within or beyond a set distance from a geographic location.

Internally, Android may use different Location Providers depending on how close you are to the outside edge of your target area. This allows the power use and cost to be minimized when the alert is unlikely to be fired based on your distance from the interface.

To set a proximity alert for a given coverage area, select the center point (using longitude and latitude values), a radius around that point, and an expiry time-out for the alert. The alert will fire if the device crosses over that boundary, both when it moves within the radius and when it moves beyond it.

When triggered, proximity alerts fire Intents, most commonly broadcast Intents. To specify the Intent to fire, you use a PendingIntent, a class that wraps an Intent in a kind of method pointer, as shown in the following code snippet:

```
Intent intent = new Intent(MY_ACTIVITY);
PendingIntent pendingIntent = PendingIntent.getBroadcast(this, -1, intent, 0);
```

The following example sets a proximity alert that never expires and triggers when the device moves within 10 meters of its target:

```
private static String TREASURE_PROXIMITY_ALERT = "com.paad.treasurealert";
private void setProximityAlert() {
    String locService = Context.LOCATION_SERVICE;
    LocationManager locationManager;
    locationManager = (LocationManager) getSystemService(locService);
    double lat = 73.147536;
    double lng = 0.510638;
    float radius = 100f; // meters
    long expiration = -1; // do not expire
    Intent intent = new Intent(TREASURE_PROXIMITY_ALERT);
    PendingIntent proximityIntent = PendingIntent.getBroadcast(this, -1,
        intent,
        0);
    locationManager.addProximityAlert(lat, lng, radius, expiration, proximityIntent);}
}
```

When the Location Manager detects that you have moved either within or beyond the specified radius, the packaged Intent will be fired with an extra keyed as LocationManager.KEY_PROXIMITY_ENTERING set to true or false accordingly.

To handle proximity alerts, you need to create a BroadcastReceiver, such as the one shown in the following snippet:

```
public class ProximityIntentReceiver extends BroadcastReceiver {
    @Override
    public void onReceive (Context context, Intent intent) {
        String key = LocationManager.KEY_PROXIMITY_ENTERING;
        Boolean entering = intent.getBooleanExtra(key, false);
        [ ... perform proximity alert actions ... ]
    }
}
```

To start listening for them, register your receiver, as shown in the snippet below:

```
IntentFilter filter = new IntentFilter(TREASURE_PROXIMITY_ALERT);
registerReceiver(new ProximityIntentReceiver(), filter);
```