

Battery Management

Renault@lrde.epita.fr



Goals and Problems



Battery is a sensitive component in mobile devices

- Think to it when writing applications
- The application must adapt to battery level
- Restrict interaction with networks



If all applications were sensitive to battery problem, the lifetime of a charge will no longer be a problem



How to build such an application?

Monitor the Battery Status



BatteryManager broadcast information about battery

- 🔊 Prefer updates when the device is charging

```
IntentFilter ifilter =
    new IntentFilter(Intent.ACTION_BATTERY_CHANGED);
Intent batteryStatus = context.registerReceiver(null, ifilter);
// Are we charging / charged?
int status =
    batteryStatus.getIntExtra(BatteryManager.EXTRA_STATUS, -1);
boolean isCharging =
    status == BatteryManager.BATTERY_STATUS_CHARGING ||
    status == BatteryManager.BATTERY_STATUS_FULL;
// How are we charging?
int chargePlug =
    batteryStatus.getIntExtra(BatteryManager.EXTRA_PLUGGED, -1);
boolean usbCharge =
    chargePlug == BatteryManager.BATTERY_PLUGGED_USB;
boolean acCharge =
    chargePlug == BatteryManager.BATTERY_PLUGGED_AC;
```

Detect the exact level of the battery

420



Exact level

```
int level = batteryStatus
    .getIntExtra(BatteryManager.EXTRA_LEVEL, -1);
int scale = batteryStatus
    .getIntExtra(BatteryManager.EXTRA_SCALE, -1);
float batteryPct = level / (float)scale;
```



Modify AndroidManifest.xml

```
<receiver android:name=".PowerConnectionReceiver">
  <intent-filter>
    <action android:name=
      "android.intent.action.ACTION_POWER_CONNECTED" />
    <action android:name=
      "android.intent.action.ACTION_POWER_DISCONNECTED" />
  </intent-filter>
</receiver>
```

Detect only important variations

421

Modify AndroidManifest.xml

```
<intent-filter>  
  <action android:name=  
    "android.intent.action.ACTION_BATTERY_LOW" />  
  <action android:name=  
    "android.intent.action.ACTION_BATTERY_OKAY" />  
</intent-filter>
```

 ACTION_BATTERY_LOW

▶ the battery is low, you should stop energy consumer activities

 ACTION_BATTERY_OKAY

▶ the battery is OK

Detecting Docks



You can trigger when the device is docked

- Car, Desk, etc. : to perform specific actions like swapping networks, silent mode, etc

```
IntentFilter ifilter =
    new IntentFilter(Intent.ACTION_DOCK_EVENT);
Intent dockStatus = context.registerReceiver(null, ifilter);
int dockState = battery.getIntExtra(EXTRA_DOCK_STATE, -1);

boolean isDocked =
    dockState != Intent.EXTRA_DOCK_STATE_UNDOCKED;
boolean isCar = dockState == EXTRA_DOCK_STATE_CAR;
boolean isDesk = dockState == EXTRA_DOCK_STATE_DESK ||
    dockState == EXTRA_DOCK_STATE_LE_DESK ||
    dockState == EXTRA_DOCK_STATE_HE_DESK;
```

Summary



Managing the battery is really important

- to have responsible behavior
- to modify the behavior of the application



Trace also connectivity to optimize battery

- CONNECTIVITY_CHANGE: helps to detect such situations
- Can be combined with connectivity manager



Some receiver can be deactivated runtime

- using PackageManager
- to avoid extra energy consumptions



