

Android (/tags/#Android)

PowerManager (/tags/#PowerManager)

Doze (/tags/#Doze)

Android电源管理之Doze模式专题系列（十二）

省电策略之电池状态

Posted by Cheson on May 16, 2017

这一篇介绍的是进入idle模式之后关于Battery State的所做的一些事情。在DeviceIdleController中接收到MSG_REPORT_IDLE_ON消息之后调用的第三个回调函数就是BatteryStats的noteDeviceIdleMode

```
mBatteryStats.noteDeviceIdleMode(true, null, Process.myUid());
```

这个方法的实现是在BatteryStatsService中

```
@Override
public void noteDeviceIdleMode(boolean enabled, String activeReason, int activeUid) {
    enforceCallingPermission();
    synchronized (mStats) {
        mStats.noteDeviceIdleModeLocked(enabled, activeReason, activeUid);
    }
}
```

先进行了对UPDATE_DEVICE_STATS权限的检查，然后实际又调用了BatteryStatsImpl中的noteDeviceIdleModeLocked方法

```

public void noteDeviceIdleModeLocked(boolean enabled, String activeReason, int activeUid) {
    // 记录系统时间
    final long elapsedTime = SystemClock.elapsedRealtime();
    final long uptime = SystemClock.uptimeMillis();
    boolean nowIdling = enabled;
    // phase 1
    if (mDeviceIdling && !enabled && activeReason == null) {
        // We don't go out of general idling mode until explicitly taken out of
        // device idle through going active or significant motion.
        nowIdling = true;
    }
    // phase 2
    if (mDeviceIdling != nowIdling) {
        mDeviceIdling = nowIdling;
        int stepState = nowIdling ? STEP_LEVEL_MODE_DEVICE_IDLE : 0;
        mModStepMode |= (mCurStepMode & STEP_LEVEL_MODE_DEVICE_IDLE) ^ stepState;
        mCurStepMode = (mCurStepMode & ~STEP_LEVEL_MODE_DEVICE_IDLE) | stepState;
        if (enabled) {
            mDeviceIdlingTimer.startRunningLocked(elapsedRealtime);
        } else {
            mDeviceIdlingTimer.stopRunningLocked(elapsedRealtime);
        }
    }
    // phase 3
    if (mDeviceIdleModeEnabled != enabled) {
        mDeviceIdleModeEnabled = enabled;
        addHistoryEventLocked(elapsedRealtime, uptime, HistoryItem.EVENT_ACTIVE,
            activeReason != null ? activeReason : "", activeUid);
        if (enabled) {
            mHistoryCur.states2 |= HistoryItem.STATE2_DEVICE_IDLE_FLAG;
            if (DEBUG_HISTORY) Slog.v(TAG, "Device idle mode enabled to: "
                + Integer.toHexString(mHistoryCur.states2));
            mDeviceIdleModeEnabledTimer.startRunningLocked(elapsedRealtime);
        } else {
            mHistoryCur.states2 &= ~HistoryItem.STATE2_DEVICE_IDLE_FLAG;
            if (DEBUG_HISTORY) Slog.v(TAG, "Device idle mode disabled to: "
                + Integer.toHexString(mHistoryCur.states2));
            mDeviceIdleModeEnabledTimer.stopRunningLocked(elapsedRealtime);
        }
        addHistoryRecordLocked(elapsedRealtime, uptime);
    }
}
}

```

这段代码分了三个if分支，就从这三段来看所做的事情。phase 1中，mDeviceIdling未初始化，第一次实例化的时候应该为false，而且在idle on的流程中，enabled为true，所以这个判断是结果为false。那么这段代码的作用是什么？考虑下接受到idle off的消息时，mDeviceIdling为true，enabled传入为false，activeReason为null，那么这个判断就走进来了，之前nowIdling的值应该被赋为了false，走进这个判断之后又改为了true。从这里的注释理解，只有准确的进入到active或者有significant motion动作之后才会正式退出idle模式。

phase 2中，在进入idle模式时，nowIdling在前面被赋值为了true，mDeviceIdling为false，所以这里的判断走进去了。先将mDeviceIdling的值赋值为true，作为当前设备处于idle模式的标示。stepState的值为STEP_LEVEL_MODE_DEVICE_IDLE（继承自父类BatteryStats，0x08），然后计算mModStepMode的值（0&0x08）^0x08|0，结果为0x08，而mCurStepMode的值为(0&~0x08)|0x08=0x08，也就是进入idle模式之后将mModStepMode和mCurStepMode这两个全局标志都赋值为了STEP_LEVEL_MODE_DEVICE_IDLE。然后判断enabled为true，开始调用mDeviceIdlingTimer.startRunningLocked(elapsedRealtime)来记录开始时间。这里的mDeviceIdlingTimer是内部类StopwatchTimer的一个示例，StopwatchTimer继承自Timer，而内部又实现了一个定时器池来管理所有定时器的生命周期。

phase 3中，和第二段类似，在idle on的时候主要让mDeviceIdleModeEnabledTimer这个定时器来计时，另外调用了addHistoryEventLocked和addHistoryRecordLocked来记录电池历史事件和历史记录，用于电量统计。

因为电池信息这部分未涉及到功耗策略部分，只是将idle模式下的耗电信息记录到BatteryStats中，所以这篇也只是对此做概要的介绍。记录电池信息的这部分在系统很多逻辑中都会走到，后续看到类似内容可以做借鉴。

PREVIOUS

ANDROID电源管理之DOZE模式专题系列（十一）
(/2017/05/12/PM_DOZE_CONN_CONTROL/)

NEXT

正信的佛教——佛教寺庙布局 (/2017/05/24/BUDDHISM_TEMPLE_ARCHITECTURE/)

FEATURED TAGS (/tags/)

- 前端 (/tags/#前端)
- Android (/tags/#Android)
- frameworks (/tags/#frameworks)
- AlarmManager (/tags/#AlarmManager)
- Performance (/tags/#Performance)
- systrace (/tags/#systrace)
- PowerManager (/tags/#PowerManager)
- Wakelock (/tags/#Wakelock)
- Guitar (/tags/#Guitar)
- 民谣 (/tags/#民谣)
- 赵雷 (/tags/#赵雷)
- Doze (/tags/#Doze)
- Android Performance Patterns (/tags/#Android Performance Patterns)

FRIENDS

待遇见志同道合的你 (https://github.com) 小明 (http://www.betterming.cn)

- (https://twitter.com/chendongqi)
- (https://www.zhihu.com/people/chendongqi)
- (http://weibo.com/chendongqi)
- (https://www.facebook.com/chendongqi)
- (https://github.com/chendongqi)
- (https://www.linkedin.com/in/firstname-lastname-idxxxx)