

## 自我挖掘 • 超越自我-Jamy Cai

[博客园](#)[首页](#)[新随笔](#)[联系](#)[订阅](#)[管理](#)

随笔 - 148 文章 - 0 评论 - 79



### 个人简介：

软件研发工程师，北京，自我挖掘

• 超越自我！

2017.4-至今：小米公司

2015.6-2017.4：搜狗公司

2013.6-2015.6：奇虎360

2012.7-2013.6：傲游公司

昵称：Jamy Cai

QQ：515581258

邮箱：Jczmdeveloer@126.com

GitHub：

<https://github.com/jczmdeveloer>

GitHub博客：<http://jczmdeveloer.github.io>

真题园网：

<http://www.zhentiyuan.com>

微博：

<http://www.weibo.com/jczmdeveloer>

昵称：Jamy Cai

园龄：5年10个月

粉丝：104

关注：1

+加关注

搜索

<input type="text"/>	<input type="button" value="找找看"/>
<input type="text"/>	<input type="button" value="谷歌搜索"/>

随笔分类(139)

Android(72)

COM/STL/WTL/ATL(26)

笔试/面试(17)

设计模式(5)

数据结构与算法(11)

数据库技术

心随之笔(8)

## Android中的Handler的机制与用法详解

### 概述：

很多android初学者对android 中的handler不是很明白，其实Google参考了Windows的消息处理机制，

在Android系统中实现了一套类似的消息处理机制。在下面介绍handler机制前，首先得了解以下几个概念：

#### 1. Message

消息，理解为线程间通讯的数据单元。例如后台线程在处理数据完毕后需要更新UI，则可发送一条包含更新信息的Message给UI线程。

#### 2. Message Queue

消息队列，用来存放通过Handler发布的消息，按照先进先出执行。

#### 3. Handler

Handler是Message的主要处理者，负责将Message添加到消息队列以及对消息队列中的Message进行处理。

#### 4. Looper

循环器，扮演Message Queue和Handler之间桥梁的角色，循环取出Message Queue里面的Message，并交付给相应的Handler进行处理。

#### 5. 线程

UI thread 通常就是main thread，而Android启动程序时会替它建立一个Message Queue。

每一个线程里可含有一个Looper对象以及一个MessageQueue数据结构。在你的应用程序里，可以定义Handler的子类别来接收Looper所送出的消息。

好了，下面是正文~

## 友情链接

[淘宝话费自助充值平台](#)[微软家园](#)[小蔡电脑助手 V3.0](#)[真题园网-www.zhenti yuan.com](#)[真题园-最全的考试真题网! 真题园考试助手, 助您圆梦! 真题园](#)

## 最新评论

1. Re:Android中的Handler的机制与用法详解

厉害啊我的哥, 虽然时隔两年多了回帖。。。刚刚入门。还有好多。请赐教

--nickName^

2. Re:Android 自定义View修炼-自定义弹幕效果View

那你这个还会覆盖弹幕算不算BUG

--Ergou

3. Re:Android开发之自定义圆形的ImageView的实现

我就想知道为什么我用这个画出来的不是一个圆, 上面是椭圆, 下面是平的, 为什么安?

--2qianqian

4. Re:Android 自定义View修炼-仿360手机卫士波浪球进度的实现

还有一个问题, 就是为什么连续快速点击会出现重合的感觉呢, 感觉若影若现, 如果停顿点击就没有相应的bug

--不加班不加班

5. Re:Android 自定义View修炼-仿360手机卫士波浪球进度的实现

为什么, 加了android:hardwareAccelerated="false", 还是显示矩形呢, 大bug啊!

--不加班不加班

## 阅读排行榜

# Android 中的 Handler 详解

## 一、Handler 是什么

1. handler 是 Android 给我们提供来更新 UI 的一套机制, 也是一套: 我们可以发送消息, 也可以通过塔来处理消息, handler 在我们的非常常见的。
2. android 在设计的时候, 就封装了一套消息创建 传递 处理机制, 的机制就没有办法更新 UI 信息, 就会抛出异常信息。

## 二、Handler 用法

1. 传递 Message。用于接受子线程发送的数据, 并用此数据配合主线程下方法:

`sendEmptyMessage(int)`

`sendMessage(Message)`

`sendMessageAtTime(Message,long)`

`sendMessageDelayed(Message,long)`

`sendMessage` 类方法, 允许你安排一个带数据的 Message 来待更新。

(1)、使用 Handler 在子线程中向 ui 线程发送一个消息进行

(2)、创建一个 Message; `Message msg = new Message();`

(3)、`handler.sendMessage(msg);`

`msg.obj=xxx;` 可以传递一个对象;

(4)、当然不一定要用 new 一个 Message, 也可以复用系统 Message `msg = handler.obtainMessage();`

2. 传递 Runnable 对象。用于通过 Handler 绑定的消息队列, 安排不序。主要有以下方法:

`post(Runnable)`

`postAtTime(Runnable,long)`

`postDelayed(Runnable,long)`

`post` 类方法允许你排列一个 Runnable 对象到主线程队列中。

3. 传递 Callback 对象。`CallBack` 用于截获 handler 发送的消息 如果: 成功 不会向下传递了。

- 1. Android开发之自定义圆形的ImageView的实现(15488)
- 2. Android开发之自定义圆角矩形图片ImageView的实现(14652)
- 3. Android开发之手势滑动（滑动手势监听）详解(12831)
- 4. Android 自定义View修炼-自定义可动画展开收缩View的实现(11262)
- 5. Android 自定义View修炼-仿360手机卫士波浪球进度的实现(6318)

评论排行榜

- 1. VC++开发垃圾文件清理软件之二：垃圾文件的扫描、显示和清理(11)
- 2. Android 自定义View修炼-Android中常见的热门标签的流式布局的实现(9)
- 3. Android 自定义View修炼-实现自定义圆形、圆角和椭圆ImageView（使用Xfermode图形渲染方法）(8)
- 4. Android 自定义View修炼-仿360手机卫士波浪球进度的实现(6)
- 5. Android 自定义View修炼-仿QQ5.0的侧滑菜单效果的实现(6)

推荐排行榜

- 1. Android 自定义View修炼-仿360手机卫士波浪球进度的实现(8)
- 2. Android 自定义View修炼-Android开发之自定义View开发及实例详解(4)
- 3. VC++开发垃圾文件清理软件之二：垃圾文件的扫描、显示和清理(4)
- 4. Android 自定义View修炼-自定义弹幕效果View(4)
- 5. Android 自定义View修炼-高仿猎豹清理大师自定义内存开口圆环比例进度View(3)

```
public Handler mHandler = new Handler(new Handler.Callback() {  
  
    @Override  
    public boolean handleMessage(Message msg) {  
        // TODO Auto-generated method stub  
        Toast.makeText(getApplicationContext(), "handleMessage 1",  
            Toast.LENGTH_SHORT).show();  
        return true;  
    }  
}) {  
    public void handleMessage(Message msg) {  
        // TODO Auto-generated method stub  
        Toast.makeText(getApplicationContext(), "handleMessage 1",  
            Toast.LENGTH_SHORT).show();  
    }  
};
```

上面的实例中，第一个有返回值的 `handleMessage` 方法是 `Callback` 的回调，如  
则不执行下面的 `handleMessage` 方法，从而达到拦截 `handler` 发送的消息的目  
回 `false`，则会继续执行 `handlerMessage` 方法。

三、 Handler 原理

3.1 Android 为什么要设计只能通过 Handler 机制更新 UI 呢？  
最根本的目的就是解决多线程并发问题  
假设在一个 Activity 当中，有多个线程去更新 UI，并且没有枷锁机制，那么  
样子问题呢，那就是更新界面混乱。如果对更新 UI 的操作进行枷锁处理的  
什么样的问题呢？那就是性能下降。Handler 通过消息队列，保证了消息顺  
序。  
鉴于以上问题的考虑，Android 给我们提供了一套更新 UI 的机制，我们只  
机制就好，所有的更新 UI 的操作都是在主线程中轮询处理。  
3.2 Handler 的原理是什么？

- (1) Handler 封装了消息的发送（主要包括消息发送给谁）  
    Looper:  
    (a) 内部包含一个消息队列也就是 `MessageQueue`，所有 Han

息都走向这个队列。

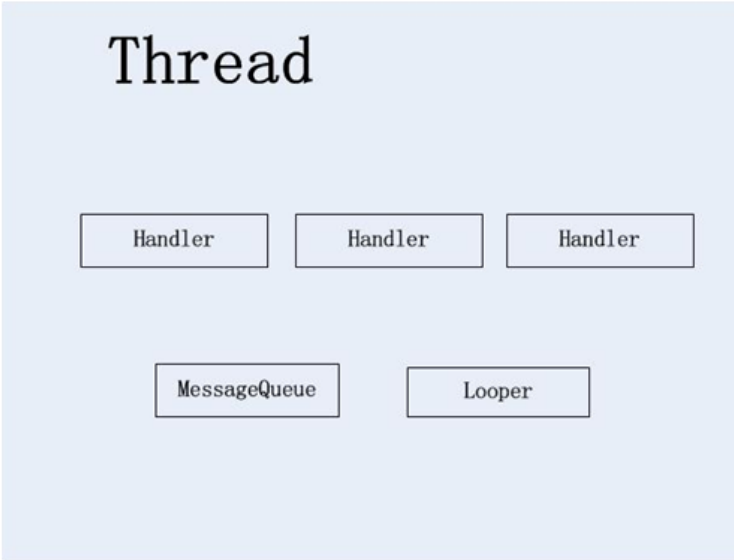
(b) `Looper.loop()`方法，就是一个 for 死循环，不断的从 `MessageQueue` 消息，如果有消息就处理消息，没有消息就阻塞。

(2) `MessageQueue`，就是一个消息队列，可以添加消息，处理消息。

(3) `Handler` 也不难，比较简单，在构造 `Handler` 时候内部会跟 `Looper` 通过 `Looper.myLooper()`获取到 `Looper`，找到了 `Looper`，也就找到了 `MessageQueue` 中发生消息，其实是向 `MessageQueue` 队列中发生消息。

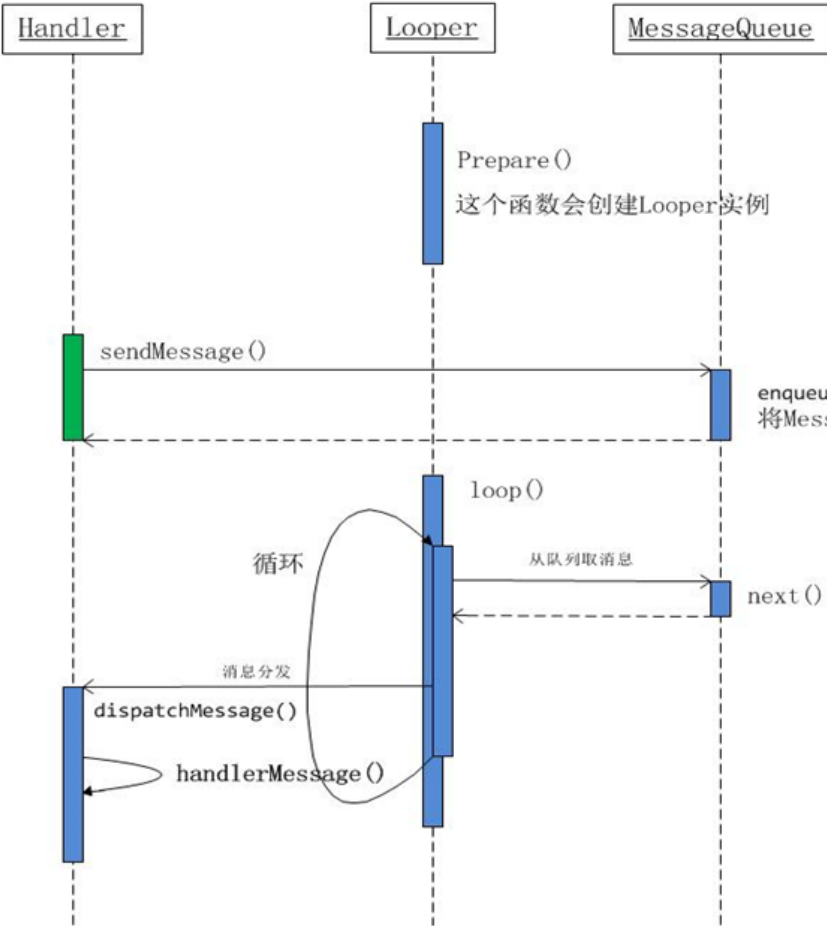
3.3 `Handler` 与 `Looper`、`MessageQueue` 的关系

总结: `handler` 负责发送消息, `Looper` 负责接收 `Handler` 发送的消息, 并传给 `handler` 自己, `MessageQueue` 就是一个存储消息的容器



如上图所示：一个线程中只有一个 `Looper` 实例，一个 `MessageQueue` 实例，多个 `Handler` 实例。

下图示意了 `Handler`、`MessageQueue`、`Looper` 之间是如何协作的。



下图是一个“我要上厕所”的形象图解：



## 四、 Handler 与子线程

### 1. 自定义与线程相关的 Handler:

```
class MyThread extends Thread{
    public Handler handler;
    @Override
    public void run() {
        Looper.prepare();//new 一个Looper对象
        handler = new Handler(){//拿到当前线程的Looper对象
            @Override
            public void handleMessage(Message msg) {
                // TODO Auto-generated method stub
                System.out.println("current thread:"+Thread.currentThread().getName());
            }
        };
        Looper.loop();//开始死循环处理消息
    }
}
```

一般 UI 主线程中不要执行一些耗时的操作，这样就可以通过子线程消息来处理。

### 2. HandlerThread 是什么

HandlerThread 继承于 Thread, 所以它本质就是个 Thread。与普通 Thread 的差别它有个 Looper 成员变量。这个 Looper 其实就是对消息队列以及队列处理逻辑。简单说就是 消息队列+消息循环。

当我们需要一个工作者线程，而不是把它当作一次性消耗品，用过即废弃的使用它。

```
private Handler mHandler = null;
private HandlerThread mHandlerThread = null;

private void sentRunnableToWorker(Runnable run) {

    if (null == mHandlerThread) {
        mHandlerThread = new HandlerThread("Worker");

        // 给工作者线程低优先级
        mHandlerThread.setPriority(Thread.MIN_PRIORITY);
        mHandlerThread.start();
    }

    if (null == mHandler)
        mHandler = new Handler(mHandlerThread.getLooper());

    mHandler.post(run);
}
```



## 五、 主线程与子线程之间的信息交互

```
//创建主线程的handler
private Handler mHandler = new Handler(){
    public void handleMessage(Message msg) {
        Message message = new Message();
        System.out.println("main Handler");
        //向子线程发送消息
        threadHandler.sendMessageDelayed(message, 1000);
    }
};
//创建子线程的handler
private Handler threadHandler;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);

    HandlerThread thread = new HandlerThread("handler");
    //创建子线程的handler
    threadHandler = new Handler(thread.getLooper()){
        public void handleMessage(Message msg) {
            Message message = new Message();
            //向主线程发送消息
            mHandler.sendMessageDelayed(message, 1000);
        }
    };
}
```

## 六、 Android 中更新 UI 的几种方式:

Android 中更新 UI 的 4 种方式: 1、[runOnUiThread](#) 2、handler 的 [post](#) 3、[sendMessage](#) 4、View 自身的 [post](#)

真题园网: <http://www.zhentiyuan.com>

分类: [Android](#)

好文要顶

关注我

收藏该文



Jamy Cai

关注 - 1

粉丝 - 104

±加关注

1

0

« 上一篇: [Android 自定义View修炼-Android 实现自定义的卫星式菜单\(弧形菜单\) View](#)

» 下一篇: [Android 自定义View修炼-Android中常见的热门标签的流式布局的实现](#)

posted @ 2015-04-08 18:02 Jamy Cai 阅读(5713) 评论(3) 编辑 收藏

### 评论列表

#1楼 2015-04-28 17:41 \_\_. 修硕

很厉害，是否收徒？ - -

支持(0) 反对(0)

---

#2楼[楼主 ] 2015-04-28 17:45 Jamy Cai

@ 之然。

我不厉害哦，很一般的码农哈，希望博文对你有所帮助

支持(0) 反对(0)

---

#3楼 2017-06-07 09:49 nickName^

厉害啊我的哥，虽然时隔两年多了回帖。。。刚刚入门。还有好多。请赐教

支持(0) 反对(0)

---

[刷新评论](#) [刷新页面](#) [返回顶部](#)

**注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，访问网站首页。**

#### 最新IT新闻:

- 高通计划帮助厂商更快打造和发售智能音响
- Windows CShell截图再爆：和桌面端一样调整动态磁贴
- Uber采纳的10大改革要点：分担CEO职责 禁止上下级谈恋爱
- 《我的世界》夏季重大更新：将支持iOS、安卓和PC玩家联机
- 支持中文！《超级马里奥》续作来了 开放地图
- » 更多新闻...

#### 最新知识库文章:

- 小printf的故事：什么是真正的程序员？
- 程序员的工作、学习与绩效
- 软件开发为什么很难
- 唱吧DevOps的落地，微服务CI/CD的范本技术解读
- 程序员，如何从平庸走向理想？
- » 更多知识库文章...

---

Copyright ©2017 Jamy Cai