**public class** Handler {

**public void** dispatchMessage(Message msg) {...}

**mHandler**.sendMessage(**new** Message());

....辗转会调用到下面的方法

**private boolean** enqueueMessage(MessageQueue queue, Message msg, **long** uptimeMillis) {  
 msg.target = **this**;//this is Handler

......  
 **return** queue.enqueueMessage(msg, uptimeMillis);  
}

**public final class** Looper {

**static final** ThreadLocal<Looper> ***sThreadLocal*** = **new** ThreadLocal<Looper>();

**final** MessageQueue **mQueue**;

**final** Thread **mThread**;

**public static void** prepare() {  
 *prepare*(**true**);  
}  
**private static void** prepare(**boolean** quitAllowed) {  
 ***sThreadLocal***.set(**new** Looper(quitAllowed));  
}

**private** Looper(**boolean** quitAllowed) {  
 **mQueue** = **new** MessageQueue(quitAllowed);  
 **mThread** = Thread.*currentThread*();  
}

**public static void** loop() {  
 **final** Looper me = *myLooper*();  
 **final** MessageQueue queue = me.mQueue;  
 **for** (;;) {  
 Message msg = queue.next(); *// might block*  
 **try** {

//msg.target is Handler  
 msg.target.dispatchMessage(msg);  
 } **finally** { }

**public static** @Nullable Looper myLooper() {  
 **return *sThreadLocal***.get();  
}

**public static void** main(String[] args) {  
 ....  
 *//创建Looper和MessageQueue对象，用于处理主线程的消息* Looper.*prepareMainLooper*();  
 *//创建ActivityThread对象* ActivityThread thread = **new** ActivityThread();  
 *//建立Binder通道 (创建新线程)* thread.attach(**false**);  
  
 Looper.*loop*(); *//消息循环运行* **throw new** RuntimeException(**"Main thread loop unexpectedly exited"**);  
}