



移动平台应用软件开发

多线程

主讲：张齐勋

zhangqx@ss.pku.edu.cn

《移动平台应用软件开发》课程建设小组

北京大学

二零一五年



北京大学



Android 中的多线程

- 主线程 (Main Thread)
 - 主要负责处理与UI相关的事件，如：用户的按键事件，用户接触屏幕的事件以及屏幕绘图事件，并把相关的事件分发到对应的组件进行处理。所以主线程通常又被叫做UI线程。
- 子线程
 - 负责除UI之外较费时的操作，如从网络上下载数据，或者访问数据库



示例：ANR(Application is not responding)

- 运行在主线程里的任何方法都尽可能少做事情。
- 可能会产生阻塞的工作不要放在主线程中。



清华大学

线程的基本用法

```
class MyThread extends Thread{  
    @Override  
    public void run() {  
        Log.d("myapp", "mythread is running...");  
    }  
}
```

```
new MyThread().start();
```

```
class MyThread implements Runnable{  
    @Override  
    public void run() {  
        Log.d("myapp", "mythread2 is running.....");  
    }  
}
```

```
MyThread mythread = new MyThread();  
new Thread(mythread).start();
```



北京大學



子线程中更新UI

- 只有UI线程才能直接操作UI控件。
- Android提供一种 **异步消息处理机制**。涉及以下四个部分：
 - Handler;
 - Message;
 - MessageQueue;
 - Looper;



代码示例

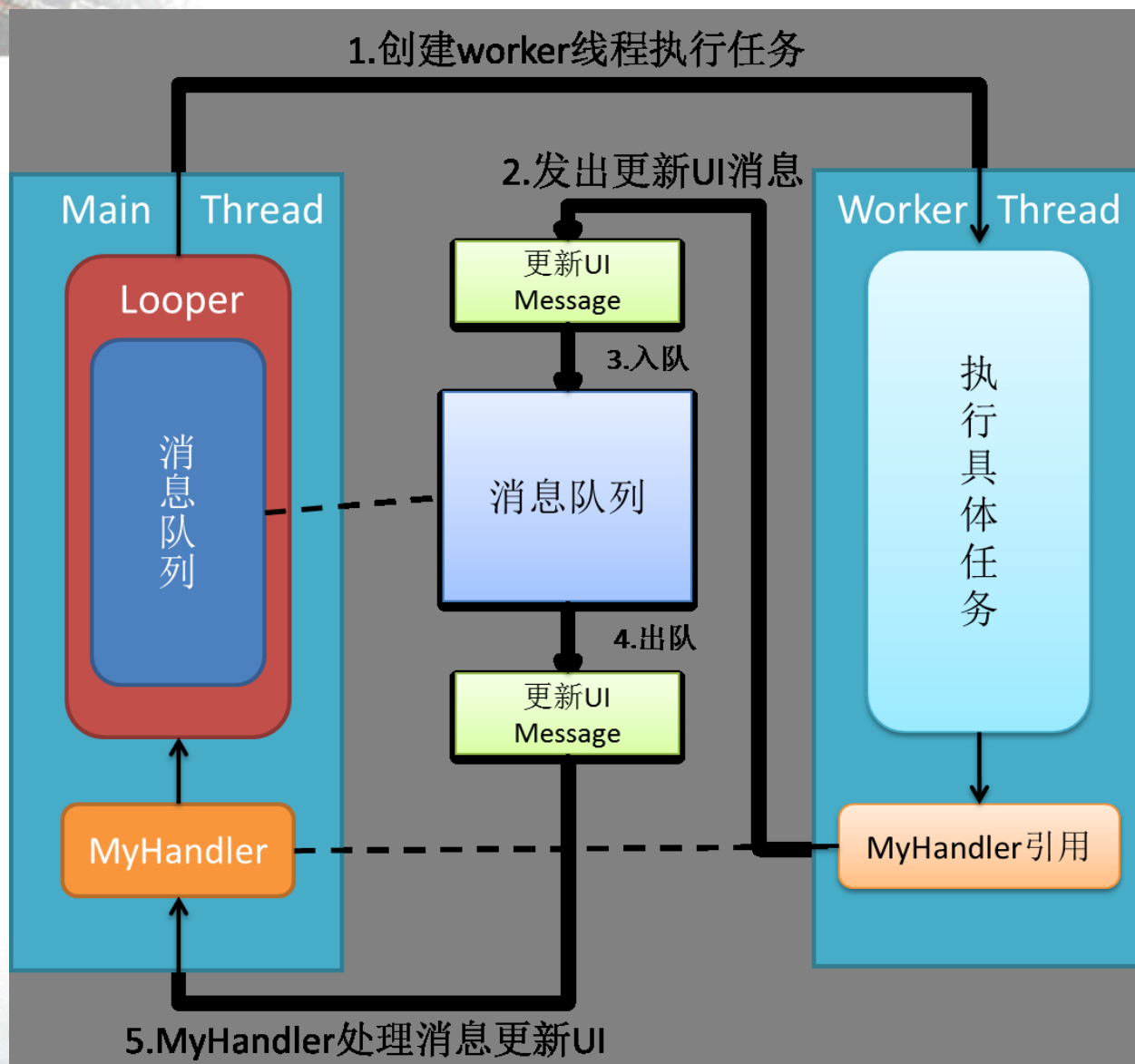
主线程增加Handler:

```
handler = new Handler() {  
    @Override  
    public void handleMessage(Message msg) {  
        switch (msg.what) {  
            case 0:  
                labelTimer.setText("逝去了 " + msg.obj + " 秒");  
        }  
    }  
};
```

子线程的循环体中增加如下语句:

```
Message msg = new Message();  
msg.obj = timer;  
msg.what = 0;  
handler.sendMessage(msg);
```







思考

- 在天气预报项目中，如何将获取的网络数据显示在UI界面上？



北京大学



其他Android多线程机制

- Android提供一个工具类**AsyncTask**。
- 实现AsyncTask类中定义的下面一个或几个方法。
 - **onPreExecute()** 开始执行前的准备工作；
 - **doInBackground(Params...)** 开始执行后台处理，可以调用publishProgress方法来更新实时的任务进度；
 - **onProgressUpdate(Progress...)**
在publishProgress方法被调用后，UI thread将调用这个方法从而在界面上展示任务的进展情况，例如通过一个进度条进行展示。
 - **onPostExecute(Result)** 执行完成后的操作，传送结果给UI 线程。





Q&A

本讲结束！



北京大学