

Design and Development of Android-Based Mobile Positioning Software

Ya Su, Yanlan Yang, Xiaojia Wu, Hua Ye

School of Automation, Southeast University, Nanjing
Email: susanna@seu.edu.cn

Received: Nov. 21st, 2012; revised: Dec. 9th, 2012; accepted: Dec. 21st, 2012

Abstract: Nowadays location-based services are being carried out in full swing at home and abroad, and extending GPS function on mobile phones has become a hot spot in mobile application development. In this paper, a mobile application is developed based on the GPS function of Android smartphone. Taking advantage of the rich set of Android APIs, position data are acquired through the android.location software package for real-time location of the phones, and transferred to the server via GPRS for real-time monitoring. The test results show that the Mobile Positioning Software runs well, and can well achieve the purpose of real-time monitoring, intervention, regulation and control as well as early warning.

Keywords: Android; Smartphone; GPS; Communication; Data Storage

基于 Android 系统的手机定位软件的设计与开发

苏 雅, 仰燕兰, 吴晓佳, 叶 桦

东南大学自动化学院, 南京
Email: susanna@seu.edu.cn

收稿日期: 2012 年 11 月 21 日; 修回日期: 2012 年 12 月 9 日; 录用日期: 2012 年 12 月 21 日

摘 要: 当前国内外基于位置的服务开展如火如荼, 在手机上拓展 GPS 功能已成为移动应用开发的一个热点。本文针对 Android 智能手机 GPS 功能进行移动应用开发, 利用 Android 丰富的 APIs, 通过 android.location 软件包获取定位的位置数据, 对手机进行实时定位, 并通过 GPRS 将位置数据传到服务器, 进行实时监控。测试应用结果表明, 手机定位软件运行良好, 能够很好地达到实时监测、干预、调控、预警人员的目的。

关键词: Android; 智能手机; GPS; 通信; 数据存储

1. 引言

随着移动通讯产业的发展, 带有 GPS 功能的手机的使用量有了大幅的增加, 利用手机进行人员定位的需求日趋强烈。Android 作为一款新型智能手机操作系统, 具有开放性好、软硬件功能扩展性强的特点, 开发基于 Android 的 GPS 应用潜力巨大^[1]。

因此, 本文利用 Android 智能手机自带的 GPS 定位功能, 获取地理坐标数据, 实时将定位数据通过 GPRS 数据网络传送至监控平台, 平台根据使用手机

的人员所到达区域, 自动分析, 做出相对应指令, 并通过 GPRS 将信息发送给手机, 从而能够很好地达到实时监控、干预、调控、预警人员的目的。

2. 整体方案设计

2.1. 软件总体设计

手机定位软件要求开机后自动后台运行, 间隔上传位置数据, 在上传位置数据时与 PC 平台采用短连接, 以接收平台发送来的信息。定位系统总体结构图

如图 1 所示。

2.2. 手机定位软件功能

手机定位软件由两部分组成，前台界面和后台服务。前台界面显示接收到的 PC 平台消息、查看定位状态以及控制后台服务的开启和关闭。后台服务实现定时定位，并实时将定位数据传送给 PC 平台，同时采用短连接等待接收 PC 平台的信息，并将信息数据存入数据库中。此外，后台服务还要定时检测 Android 智能机的网络状态、电量状态和 GPS 使能状态，以保证软件正常运行。最后，后台服务须向前台界面传递数据，以进行界面更新。手机定位软件功能结构框图如图 2 所示。

2.3. 开发环境的建立

Android 支持 Windows XP 以上的操作系统，Android 中的应用程序都是 Java 程序，首先需要 Java 环境，用以支持 Android 和 Eclipse。然后安装 Eclipse 开发工具，Eclipse 是一个基于 Java 的开发平台，它是一个框架，通过安装插件构建开发环境，Java 就是它的插件，Android 也是一组插件，利用它开发和调试运行在 Android 上的程序。接着下载 Android SDK，Android SDK 包含创建和运行 Android 应用程序所需的一切，包括设备仿真器和高级调试工具。安装 Android 的 Eclipse 插件 ADT。最后配置 Android 模拟器 AVD^[2]。

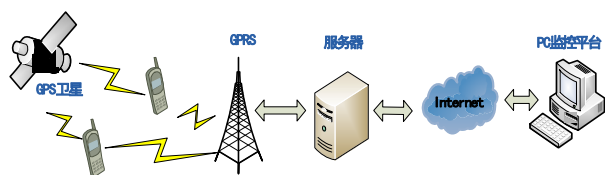


Figure 1. The overall structural diagram of the positioning system
图 1. 定位系统总体结构图

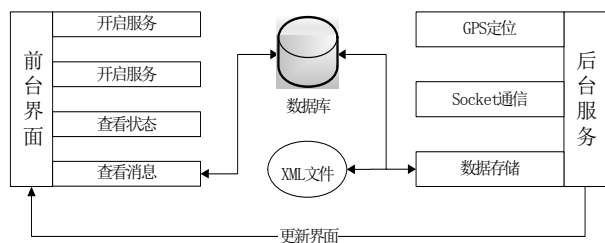


Figure 2. The diagram of function structure
图 2. 功能结构框图

3. Android 组件的使用

Android 中没有入口函数，应用程序以各种组件为基础，所有的组件都是平行的，都可以单独实例化^[3]。Android 组件是 Android 应用程序的核心，设计中运用到的 Android 组件包括：

1) 屏幕组件——Activity

Activity 是 Android 的核心类，可以理解为一个屏幕，显示给用户的界面^[3]。一个 Android 应用程序可以认为由多个 Activity 的交互组成的，多个 Activity 之间可以进行相互跳转，不同的 Activity 实现不同的功能。

2) 服务组件——Service

Service 是 Android 的服务组件，没有用户界面，在后台运行，对用户完全透明。

本设计需要手机在开机后自动后台运行，因此手机软件的定位和通信功能在 Service 中设计实现。

3) 广播接收组件——BroadcastReceiver

BroadcastReceiver 组件的功能就是用来接收广播并且做出回应^[3]。该组件和 Service 组件类似，没有用户接口，但是广播接收者能启动 Activity 来对接收到的广播做出回应，一个应用程序可以有多个广播接收者。

在收到开机广播后，可通过调用 Context.startService() 语句启动服务，实现开机自启动服务。在收到关机广播时，向 PC 平台发送关机报警。此外，Service 向 Activity 传递数据也可借助于 Broadcast Receiver 实现。

Activity、Service 和 BroadcastReceiver 组件在设计中相互之间的联系如图 3 所示。

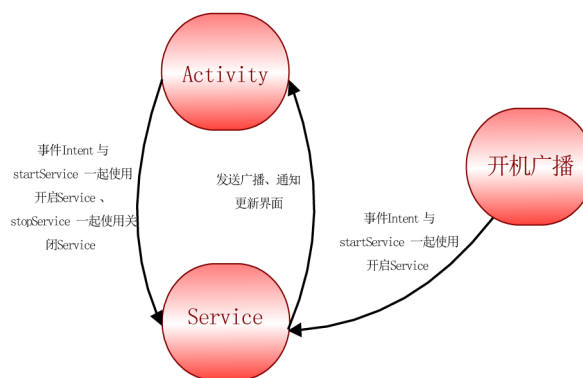


Figure 3. The diagram of the android components used
图 3. Android 组件使用的联系图

4. 软件详细设计

4.1. 定位实现

4.1.1. GPS 定位前提

使用 Android 智能机自带的 GPS，首先要在手机的“设置”中“位置服务”中勾选“GPS 卫星”。此外为了能够让应用程序获取定位服务的能力，程序中还需要添加如下两种定位服务权限：

```
<uses-permissionandroid:name = "android.permission.ACCESS_FINE_LOCATION"/>
```

```
<uses-permissionandroid:name = "android.permission.ACCESS_COARSE_LOCATION"/>
```

4.1.2. Android 定位 API

Android 关于地理定位的 API 主要集中在 android.location 软件包内^[4]。本设计基于 GPS 定位实现定位功能，两个重要的类是 LocationManager 类和 Location Provider 类^[5]。LocationManager 类是定位服务最主要的类，负责定位服务管理，包括获取定位数据、读取 GPS 状态等，Location Manager 的对象不需要实例化，而是通过获取系统服务来实现的^[2]，用到 Location Manager 类中的方法主要如表 1 所示。LocationProvider 类用来描述位置提供者，设置位置提供者的一些属性，通过 Criteria 类来为 LocationProvider 类设置条件，获得合适的 LocationProvider。

4.1.3. 编程实现

Android 的线程分为两类：主线程和子线程。当

Table 1. The functions about positioning
表 1. 定位有关的方法

方法名称	功能说明
isProviderEnable 方法	获取位置提供者是否使能。
getLastKnownLocation 方法	用于获取 Android 系统中最新保存的位置信息，其中包含了经纬度等位置数据，程序中通过解析该返回值便可以获取位置信息，通过该方法主动地获取 GPS 定位的位置信息，但该数据并一定代表实时的定位数据。
requestLocationUpdates 方法	用于向 Android 系统请求位置更新，当用户的位置更新时将及时获取最新的定位数据，该方法获得位置更新的方式可以有多种，如监听器、Intent 等。
removeUpdates 方法	用于移除 Android 系统的位置更新请求，这样，当用户的位置更新时将不再获取最新的定位数据。

一个应用程序第一次启动的时候，Android 会启动一个 LINUX 进程和一个主线程，默认情况下，所有该程序的组件都将在该进程和线程中运行^[6]。主线程主要负责处理与 UI 相关的事件，并把相关的事件分发到对应的组件进行处理，如果在主线程中占用太长的时间的话，就会引发 ANR(应用程序无响应)对话框，此时需要把一些耗时操作如访问网络或数据库等放到一个新的子线程中进行^[6]，并通过使用 Android 提供 Handler 类实现主线程和子线程之间数据的交换。

手机定位软件采用多线程技术，在主线程中创建一个 handler 对象用 handleMessage 方法接收子线程中的消息和用 postDelayed 方法发 Runnable 对象到主线程中，通过重载 Runnable 对象的 run 方法实现定时定位，在子线程中创建 Socket 与 PC 平台进行通信。定时定位程序流程图如图 4 所示。

4.2. Socket 通信

4.2.1. Socket 简介

Socket 通常也称作“套接字”，用于描述 IP 地址和端口，是一个通信链的句柄。应用程序通常通过“套接字”向网络发出请求或者应答网络请求。为了让应用程序拥有联网权限，程序中还需要添加如下权限：

```
<uses-permissionandroid:name = "android.permission.INTERNET"/>
```

4.2.2. 通信协议

GPS 定位软件通过 Socket 建立 TCP 连接后，与 PC 平台进行通信，自定义通信协议如表 2 所示。

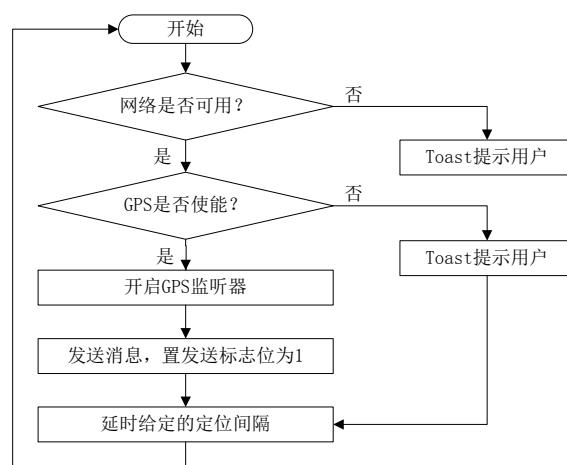


Figure 4. The flow chart of timing positioning
图 4. 定时定位程序流程图

Table 2. Communication protocols
表 2. 通信协议

(a) 终端→服务器

信息头 2 字节	内容长度 1 字节	保留位 2 字节	终端 ID 8 字节	序列号 2 字节	协议号 1 字节	信息内容 24 字节	结束标志 2 字节
68H68H	25H	00H00H			10H		0DH0AH

(b) 服务器→终端

信息头 2 字节	间隔时间 3 字节	终端 ID 8 字节	协议号 1 字节	信息内容 最多 140 字节	结束标志 2 字节
54H68H			1BH		0DH0AH

4.2.3. 编程实现

GPS 监听器监听位置变化需要一定的时间，在主线程中不能用 Sleep 方法延时，会导致整个进程处于休眠状态，因此，在主线程中打开 GPS 监听后，发送消息置发送标志为 1，当网络通信子线程检测到发送标志为 1 时，延时一定的监听时间后，发送消息关闭 GPS 监听器，然后进行 Socket 通信，程序流程图如图 5 所示。

4.3. 数据存储

为了能够查看 PC 平台发送的所有信息，并根据使用人员的需要进行单个或多个删除信息，手机定位软件要对接收到的消息进行存储，此外，还需要存储一些标志位的状态和软件的相关配置。

Android 提供了多种数据存储方式，如 Shared Preferences、文件存储、SQLite 数据库等，每一种存储方式都各有特点，设计中根据它们各自的特点采用了 SharedPreferences 和 SQLite 数据库存储方式。

SharedPreferences 是 Android 中提供了一种轻量级的存储方式，用键值对的方式把简单的数据类型存储在应用程序的私有目录下的 xml 文件中，类似 Windows 中的 ini 文件，Web 中的 Cookie，用它来存储一些标志位的状态和手机定位软件的相关配置。

Android 提供 SQLite 数据库用于大量数据的存储，其最大的特点就是轻量，几乎不占用资源，具有很强的独立性、隔离性、跨平台性和安全性，并且支持多种编程语言，只有表格一种数据结构，是一个非常优秀的嵌入式数据库^[3]。Android 系统主要通过 SQLiteDatabase 类、SQLiteOpenHelper 类和 Cursor 接口来对数据库进行操作，可以完成数据的增加、删除、

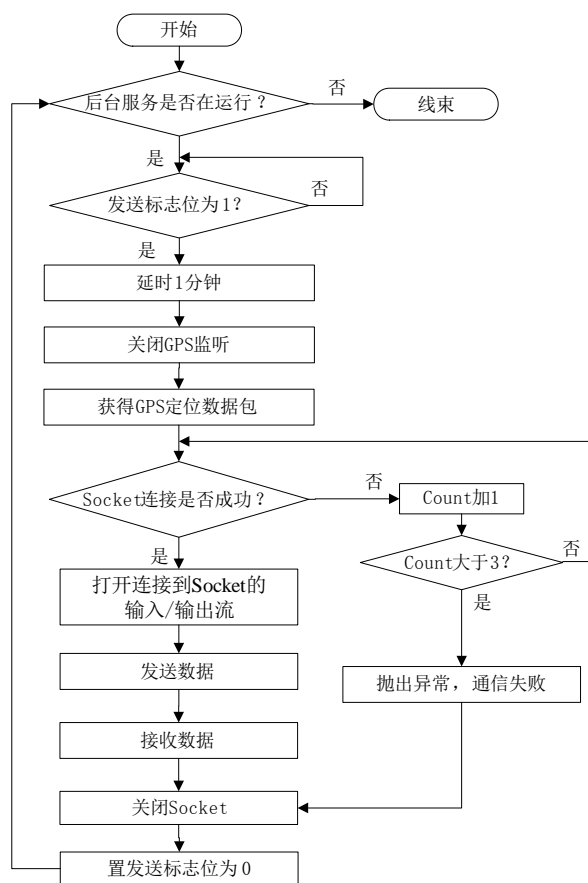


Figure 5. The flow chart of socket communication thread
图 5. Socket 通信线程流程图

更新和查找等基本操作，同时也可以执行大多数的 SQL 语句^[2]。设计中用 SQLite 数据库来存储 PC 平台发送的信息。

4.4. Android 睡眠唤醒

当手机处于待机状态时，后台服务工作异常。一般手机待机时，Android 手机的 Application 处理器、

显示屏、网络均进入休眠状态，这时 Android 中应用程序的代码也会停止执行，Android 为了确保应用程序中关键代码的正确执行，提供了 Wake Lock 的 API，使得应用程序有权限通过代码阻止 Application 处理器进入休眠状态。

通过在后台服务生命周期的开启阶段中申请 WakeLock 让进程持续执行即使手机进入睡眠模式，在后台服务生命周期的销毁阶段中释放 WakeLock，可解决手机待机状态时后台服务工作异常的问题。

4.5. 系统回收问题

当手机内存不足时，后台服务会被 Android 系统回收。Android 系统对于内存管理有自己的一套方法，为了保障系统有序稳定的运行，系统内部会自动分配、控制程序的内存使用，当系统觉得当前的资源非常有限的时候，为了保证一些优先级高的程序能运行，就会回收一些它认为不重要的程序或者服务来释放内存，这样就能保证真正对用户有用的程序仍然再运行。

在 Android 系统中，进程分为高优先级、中优先级和低优先级三类，按照优先级由高到低的顺序，可以归类如下^[2]。

- 高优先级：前台进程。
- 中优先级：可见进程、服务进程。
- 低优先级：后台进程、空进程。

由此可见前台进程的优先级最高，相比被系统回收的概率最小。因此通过将后台服务设置成前台状态的方法，减小了被系统自动回收的概率，保证手机定位软件长时间正常的运行。

5. 软件运行结果

基于 Android 操作系统进行的 GPS 应用设计，充分利用 Android 通用框架和与设备无关的应用程序开发平台，实现了手机定位软件总体架构。在软件的设计中充分考虑了 Android 系统的易移植、高效率等优点，集中使用了 Android 系统各接口 API 模块，使系统从开发效率和运行效率上都有较大的提升。测试结果表明本应用程序能较好的实现 GPS 定位功能。

图 6 为应用软件主界面的窗口，点击查看状态跳转到 GPS 定位状态窗口，如图 7 所示，点击查看消息跳转到来处平台的消息窗口，如图 8 所示。



Figure 6. The main interface
图 6. 主界面



Figure 7. Positioning state
图 7. 定位状态面



Figure 8. Messages
图 8. 信息

6. 结束语

本文利用 Android 手机自身的 GPS 模块进行定位。根据通信协议，对定位信息进行数据封装，实时将位置数据传送到服务器管理平台，并保持一定的连接时间等待接收服务器管理平台下发的信息，方便管理人员对使用手机的相关人员进行实时监控与调度，提高了工作效率和运营效益。测试应用结果表明，定位软件运行良好，能够很好地达到实时监控、干预、调控、预警人员的目的。

参考文献 (References)

- [1] 公磊. 基于 Android 的 GPS 测量系统开发[D]. 南昌大学, 2008.
- [2] 杨明羽. Android 语法范例参考大全[M]. 北京: 电子工业出版社, 2012.
- [3] 裴佳迪, 马超, 孙仁贵. Android 应用开发全程实录[M]. 北京: 人民邮电出版社, 2012.
- [4] 代敏. 基于 Android 平台下手机定位程序的设计及实现[J]. 计算机与数字工程, 2012, 40(4): 143-145.
- [5] 祝洪娇. 基于 Android 平台的位置服务系统的设计与实现[D]. 北京交通大学, 2011.
- [6] 汪源, 刘芳. Android 的进程与单线程模型浅析[J]. 仪器仪表用户, 2011, 5(18): 77-78.