**详细设计**

# 模块设计

Android事件助手从开发的角度分为5个模块。它们分别是联系人、事件、共享、备份与恢复、日历。每个模块的大部分开发代码是严格内聚的，当需要调用其它模块或者被其它模块调用时，采用设计好的标准接口。因此在开发过程中可以尽可能的减少之间的依赖，做到同步开发，同步测试。个别模块之间存在相互调用。它们是：事件模块需要用到联系人模块、事件需要用到日历模块、共享需要用到事件模块、备份与恢复需要用到事件模块。总的开发模块图如下：



图1：开发模块图

模块依赖详细说明（包括模块内）：

1. 联系人信息依赖群组
2. 联系人生日编辑依赖联系人信息
3. 事件创建需要调用联系人信息
4. 事件搜索依赖事件信息
5. 事件提醒服务依赖事件创建
6. WIFI共享，蓝牙共享依赖事件信息
7. 备份管理依赖事件信息
8. 恢复管理依赖备份管理

# UI设计

## 2.1联系人UI

联系人用户界面设计中包括四张图。分别是联系人详情、联系人列表、联系人编辑、联系人选择。

下图是联系人详情界面。位于最顶端的是用户头像和姓名，下面是生日显示，生日下面是用户输入的备注信息。用户可以调出optionMenu来进行编辑。

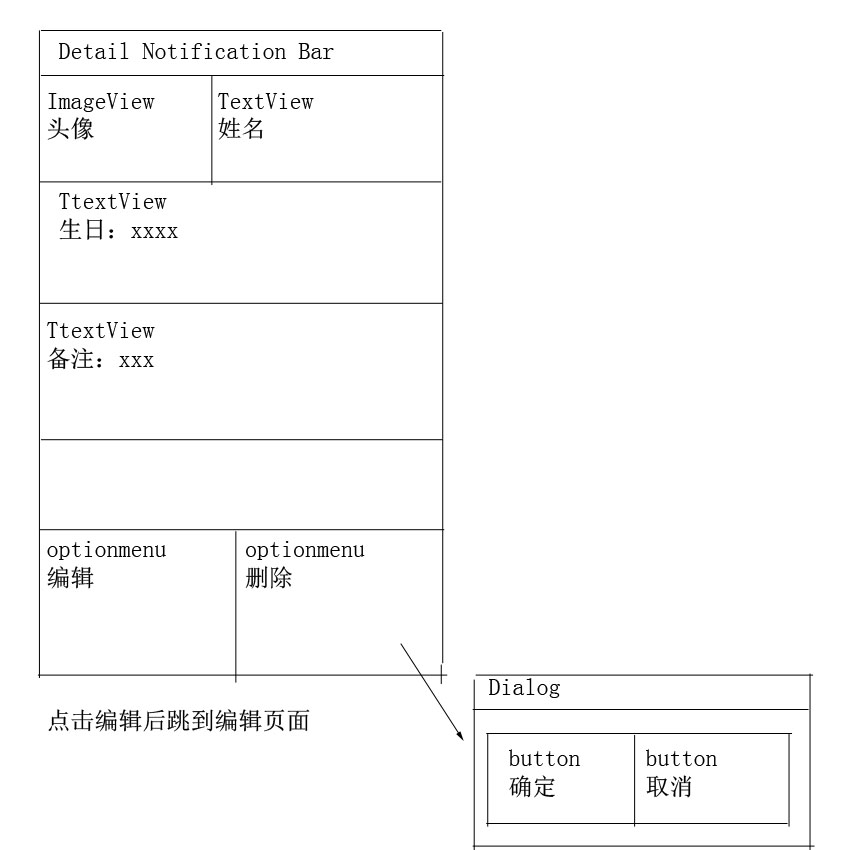


图2：联系人详情

下图是联系人编辑用户界面。位于顶端的头像和姓名是不可以更改的。下方的生日可以点击按钮进行选择。其它信息皆不可更改。最后用户可以点击最下方的取消按钮退回上一个界面或者点击保存按钮保存信息。

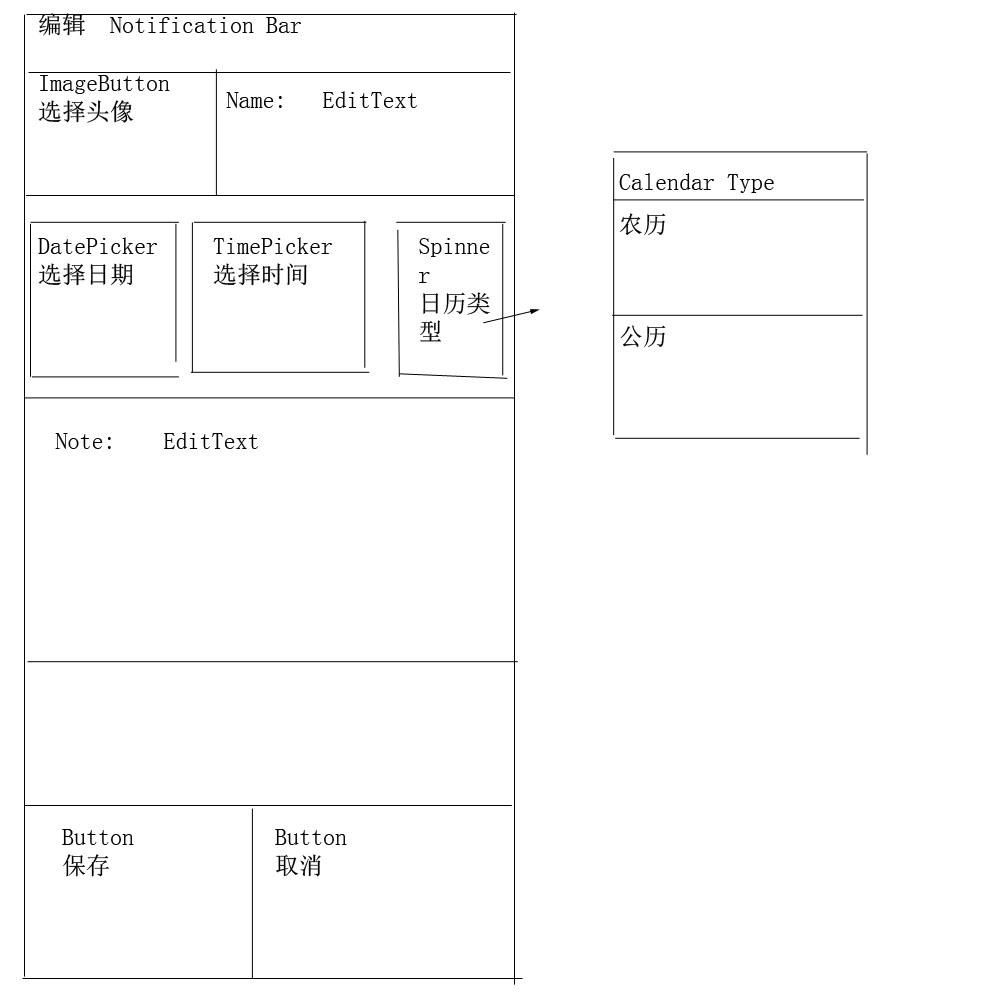


图3 联系人编辑

下图是联系人列表界面。这个列表包括两部分。全部（Button）是一部分，Button下面又是一部分。全部按钮点击后，下面的相应该组的联系人全部消失，出现群组选择列表。用户在群组选择列表中选择一个组后，群组选择列表消失，Button的名称变为相对于的群组名称。联系人列表中显示该组中的所有联系人。

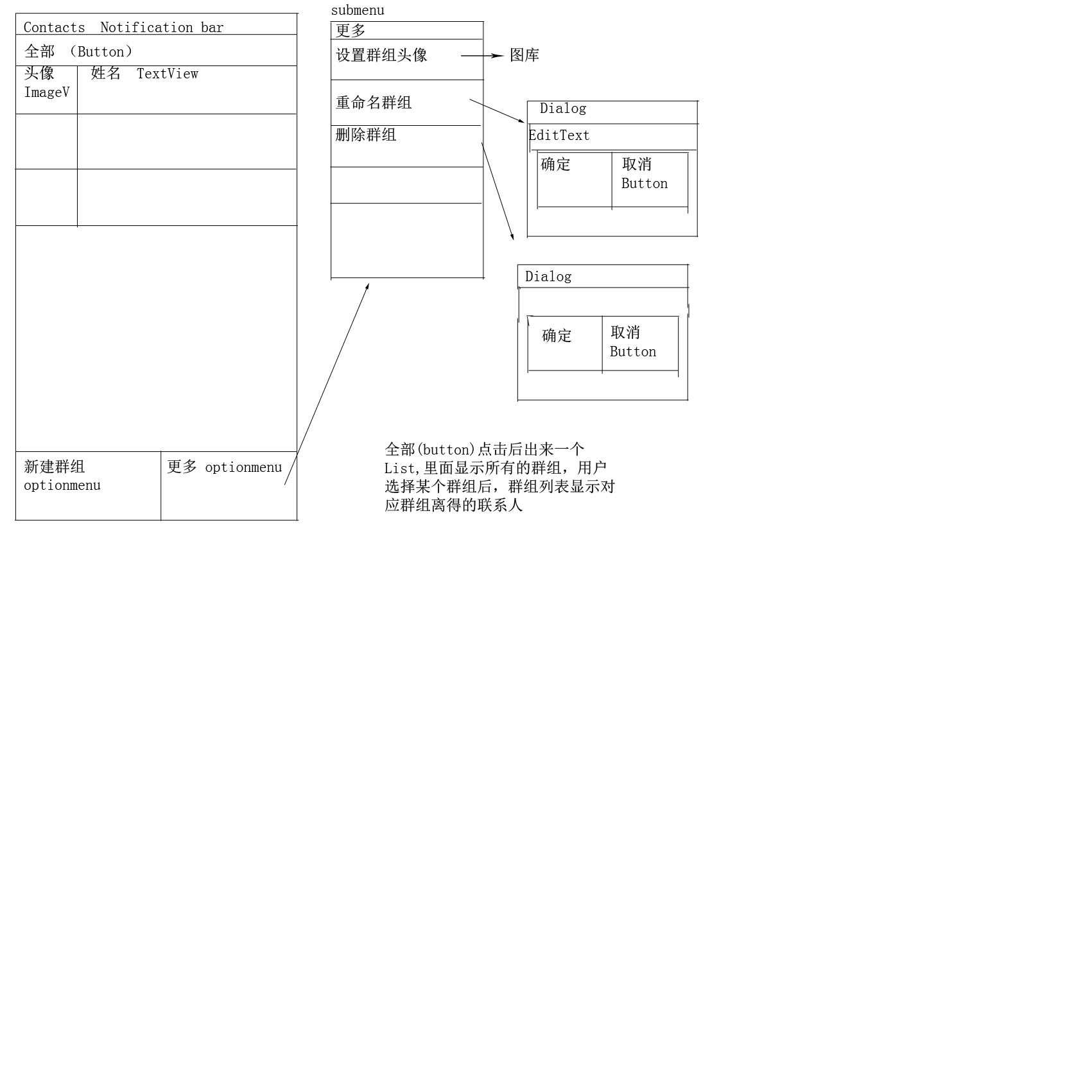


图4 联系人列表

## 2.2事件UI

事件用户界面设计中包括三张图。分别是事件列表、事件详细信息、编辑（新增）事件。

下图是事件列表界面。首先列出当天的节日和联系人生日，然后列出用户添加的事件，选中事件则会出现删除按钮，方便用户删除事件。用户点击某个事件则跳转到事件详细信息界面。用户可以调出optionMenu来新增事件。



图5 事件列表

下图是事件详细信息界面。依次列出事件各项信息，用户点击编辑按钮则跳转到编辑事件界面。



图6 事件详细信息

下图是编辑（新增）事件界面。依次列出事件各项信息，用户可以修改各项内容然后保存。



图7 编辑（新增）事件

## 2.3共享UI

## 2.4备份与恢复UI

## 2.5日历UI

下图是日历的用户界面。最顶端的是月份，下方是星期和对应的日期。月份是可以由用户点击选择的，另外一种跳转月份的功能是用手指在触摸屏上滑动，向左滑动月份向前跳转，向右滑动月份向后跳转。每一个日期都是可以点击的。如果该日期内有事件，则该日期所在的框内会有一个红色的标志来提醒用户。用户点击相应的日期，会跳转到事件列表用户界面。

同时该用户界面有optionsMenu。其中的功能有日期快速跳转和事件搜索。



图X: 日历

## 2.6设置UI

下图是事件助手设置的用户界面。里面有若干设置选项。你喜欢的提醒方式代表事件时间到后该如何提醒用户，方式有响铃，振动，通知栏。默认前日提醒是在用户设置了前日提醒后系统设置的默认天数和每天提醒次数，用户给以手动更改。自动备份开关可以设置软件是否自动备份事件等信息。自动备份时间代表每隔多少时间系统自动备份一次。备份位置可选内存和SD卡。



## 2.7帮助UI

# 数据库设计



实体：

Event，Event\_Contact，RawContact，其中RawContact是Android系统自带表，这里仅表示实体关系，其它数据未标出。

联系：

Event\_Contact和Event是多对一关系，Event\_Contact和RawContact是多对一关系。

属性：

Event：

内容

开始时间

结束时间

发生地点

备注

当日提醒时间

提前几天提醒

每天提醒的次数

提醒方式（响铃、振动）

Event\_Contact：

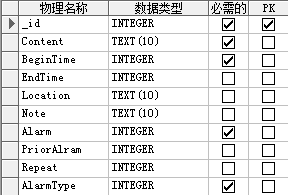
事件ID

联系人ID

RawContact：

系统自带联系人属性

Event



Event\_Contact



# 类结构设计

## 联系人类结构

下图是联系人模块类图。图中包括了最重要的相关类，类中标出了重要的成员变量和方法。类与类之间标出了依赖和继承等关系。

GroupUtils, ContactUtils, Group, Contact位于左边的包内ustc.sse.assistant.contact.data

ContactSelection, ContactSelection, ContactEdit, ContactDetail,ContactIndividualViewBinder位于右边的包内ustc.sse.assistant.contact。

下面是类的详细内容表格：

属性表格

|  |  |
| --- | --- |
| **Class: Group** | |
| **Attribute** | **Attribute Description** |
| groupId | 代表该组的ID号 |
| note | 该组的备注 |
| SummaryCount | 该组的人数 |
| SummaryCountWithPhone | 该有有电话的人数 |

属性表格

|  |  |
| --- | --- |
| **Class: Contact** | |
| **Attribute** | **Attribute Description** |
| contacted | 代表该联系人的ID号 |
| birthday | 该联系人的生日 |
| displayedName | 该联系人的主要姓名 |
| eventType | 事件类型这里是生日 |
| note | 联系人备注 |
| Phone | 联系人电话 |
| PhoneType | 电话类型(家庭，手机等) |
| Photo | 联系人照片 |
| PhotoId | 联系人照片的ID号 |

属性表格：

|  |  |
| --- | --- |
| **Class: ContactUtils** | |
| **Attribute** | **Attribute Description** |
| Activity | 代表使用该类的activity |

方法表格：

|  |  |
| --- | --- |
| **Class: ContactUtils** | |
| **Method** | **Method Description** |
| contactsIntoList | 将多个联系人放入一个列表 |
| contactsToGroups | 将多个联系人放入一个组 |
| listToMap | List<Contact> to List<Map<String, Object>> used in SimpleAdapter |
| getAllConactsBasicInfo | 获得所有联系人的基本信息 |
| getBirthday | 获得指定联系人的生日 |
| getContactById | 通过ID获得该联系人的所有信息 |
| getContactsBasicInfoByGroup | 获得一组内的所有联系人 |
| GetDisplayName | 获得联系人的主要名称 |
| getNote | 获得联系人备注 |
| getPhoto | 获得联系人照片 |
| getPrimaryPhone | 获得联系人主要电话 |
| getRawContactId | 获得一个contactId对应的所有rawContactId |
| updateContact | 更新联系人 |

方法表格：

|  |  |
| --- | --- |
| **Class: ContactSelection** | |
| **Method** | **Method Description** |
| getChildData | 获得相应组内的所有联系人基本信息 |
| initAdapter | 初始化adapter，用于ExpandableListView。调用getChildData |
| initAllMembersGroup | 初始化一个组，这个组称为All，其中包括所有联系人 |
| onCreate | 回调函数 |
| onStart | 回调函数，其中按顺序调用initAllMembersGroup, initAdapter |

属性表格：

|  |  |
| --- | --- |
| **Class: ContactSelection** | |
| **Attribute** | **Attribute Description** |
| contactUtils | ContactUtils的一个实例 |
| groupUtils | GroupUtils的一个实例 |

属性表格：

|  |  |
| --- | --- |
| **Class: ContactDetail** | |
| **Attribute** | **Attribute Description** |
| DEFAULT\_GROUP | 默认optionMenu组的id号 |
| EDIT\_MENU\_ITEM\_ID | optionMenu中的item\_id |
| EDIT\_MENU\_ITEM\_ORDER | optionMenu中的item\_order |
| contact | 一个联系人实例 |
| contactUtils | 一个ContactUtils实例 |

方法表格：

|  |  |
| --- | --- |
| **Class: ContactDetail** | |
| **Method** | **Method Description** |
| loadContactDetail | 读取该联系人的全部信息 |
| onCreate | 回调函数，调用loadContactDetail |
| onCreateOptionMenu | 回调函数，创建optionMenu |
| onOptionItemSelected | 回调函数，optionMenu点击后触发 |
| onResume | 回调函数 |

属性表格：

|  |  |
| --- | --- |
| **Class: ContactEdit** | |
| **Attribute** | **Attribute Description** |
| BIRTHDAY\_DATE\_PICKER | Dialog 编号 |
| DATA\_FORMAT | 格式化日期的Pattern |
| birthdayNull | Boolean变量代表生日是否为空 |
| contact | 修改的联系人实例 |
| contactUtils | 一个ContactUtils实例 |

方法表格：

|  |  |
| --- | --- |
| **Class: ContactEdit** | |
| **Method** | **Method Description** |
| createBirthdayDatePickerDialog | 创建日期选择对话框 |
| initButton | 初始化Save，Cancel按钮 |
| LoadContactInformation | 根据Intent中的ID来读取联系人信息 |
| onCreate | 回调函数，依次调用initButton, loadContactInformation |
| onCreateDialog | 创建日期选择对话框 |

属性表格：

|  |  |
| --- | --- |
| **Class: ContactList** | |
| **Attribute** | **Attribute Description** |
| DEFAULT\_GROUP | 群组编号表示默认的群组 |
| NONE\_GROUP\_ID | 群组编号表示尚未选择任何组 |
| Tag | 标签 |
| contactListView | 联系人列表 |
| contactUtils | 一个ContactUtils实例 |
| groupListView | 群组列表 |
| groupUtils | 一个groupUtils实例 |
| lastSelectedGroup | 上次选择的群组编号 |
| sharedPreference | 属于该activity的sharedPreference实例 |

方法表格：

|  |  |
| --- | --- |
| **Class: ContactList** | |
| **Method** | **Method Description** |
| generateContactListView | 创建联系人列表 |
| initListener | 初始化各个监听器 |
| loadLastSelectedGroup | 载入上次选择的群组编号 |
| onCreate | 回调函数，载入Layout，初始化成员变量 |
| OnPause | 回调函数，调用saveLastSelectedGroupId |
| onStart | 回调函数，依次调用  loadLastSelectedGroup，  prepareGroupListView，  prepareGroupMemberListView，  initListener |
| prepareGroupListView | 准备好群组列表 |
| prepareGroupMemberListView | 准备每个相应群组的联系人 |
| saveLastSelectedGroupId | 储存选择的群组编号 |

图X: 联系人类图

## 事件类结构

## 日历类结构

## 共享类结构

## 备份与还原类结构

## 设置类结构

## 帮助类结构