中兴Android系统界面软件设计与开发

**软件结构设计说明**

**版本：0.1**

编写： 张 翔

校对： 迟建涛

审核： 杨 帆

批准： 朱怡安

**西北工业大学－中兴通讯嵌入式系统联合开发实验室**

**2011年10月**

**目 录**

[1引言 4](#_Toc310763636)

[1.1文档标识 4](#_Toc310763637)

[1.2项目概述 4](#_Toc310763638)

[1.3文档概述 4](#_Toc310763639)

[1.4参考文档 5](#_Toc310763640)

[2 CSCI级设计决策 6](#_Toc310763641)

[3 CSCI体系结构设计 6](#_Toc310763642)

[3.1 程序（模块）划分 6](#_Toc310763643)

[3.2 程序（模块）层次结构关系 7](#_Toc310763644)

[4 CSCI 部件设计 8](#_Toc310763645)

[4.1 Launcher 8](#_Toc310763646)

[4.1.1 功能描述 8](#_Toc310763647)

[4.1.2 部件组成 8](#_Toc310763648)

[4.1.3 接口设计 11](#_Toc310763649)

[4.2 动态壁纸 13](#_Toc310763650)

[4.2.1 星光漂移 13](#_Toc310763651)

[4.2.2 实时时间天气 15](#_Toc310763652)

[4.3 桌面Widget 19](#_Toc310763653)

[4.3.1 三维时间天气 19](#_Toc310763654)

[4.3.2 网络书签 22](#_Toc310763655)

[4.3.3 联系人 24](#_Toc310763656)

[4.3.4 社交网络 26](#_Toc310763657)

[4.3.5 微博 27](#_Toc310763658)

[4.4 锁屏 29](#_Toc310763659)

[4.4.1 功能描述 29](#_Toc310763660)

[4.4.2部件组成 30](#_Toc310763661)

[4.4.3 接口设计 30](#_Toc310763662)

[4.5 任务管理器 31](#_Toc310763663)

[4.5.1 功能描述 31](#_Toc310763664)

[4.5.2部件组成 32](#_Toc310763665)

[4.5.3 接口设计 33](#_Toc310763666)

# 1引言

## 1.1文档标识

中文名称：《软件结构设计说明》。

英文名称：“Software Architecture Design Description（SADD）”。

文档版本：“0.1”。

文档编号：“SSM-ZTE-AndroidUI-SADD-0.1”。

## 1.2项目概述

本文档适用于“中兴Android系统界面软件设计与开发”项目（以下简称“AndroidUI项目”）的开发过程。AndroidUI项目由中兴通讯股份有限公司（以下简称“中兴通讯”）提出，由西北工业大学－中兴通讯嵌入式系统联合开发实验室（以下简称“联合实验室”）负责实施，该项目标识号为“SSM-ZTE-AndroidUI”，其软件产品版本号为“1.0”，包括三个内部版本，分别是0.1版、0.2版和0.3版。

项目内容为：

基于Android 3.2系统版本，针对中兴通讯设计开发的平板电脑，联合实验室依据中兴通讯提出的用户需求和设计的软件界面，深度定制Android系统的人机界面及应用软件界面，完成中兴Android系统界面软件的设计与开发。

## 1.3文档概述

本文档依据国家标准[《GB/T 8567-2006计算机软件文档编制规范》](../../资料/GBT%208567-2006%20计算机软件文档编制规范.pdf)制定，主要从CSCI设计决策、CSCI体系结构设计、CSCI部件设计三个方面描述本软件的结构设计，属于技术文档，仅限于联合实验室和中兴通讯的项目相关人员阅读。

## 1.4参考文档

* [《GB/T 8567-2006计算机软件文档编制规范》](../../资料/GBT%208567-2006%20计算机软件文档编制规范.pdf)，国家标准
* [《高校合作项目要求说明书－中兴Android系统界面软件设计与开发（讨论稿）》，](../../资料/高校合作项目要求说明书－中兴Android系统界面软件设计与开发（讨论稿）.doc)中兴通讯提供

# 2 CSCI级设计决策

本章内容可参考《SSM-ZTE-AndroidUI-UIDD-0.1 用户界面设计说明》及《SSM-ZTE-AndroidUI-SFT-0.1 软件功能列表》。

# 3 CSCI体系结构设计

## 3.1 程序（模块）划分

该软件可划分为以下9个程序，分别是Launcher启动器， 星光飘移壁纸、实时时间天气壁纸、三维时间天气widget、网络书签widget、社交网络widget、微博widget、锁屏应用以及任务管理器应用。

## 3.2 程序（模块）层次结构关系

Launcher

LiveWallpaperPicker

锁屏

任务管理器

AppWidgetManager

三 维 时 间 天 气

网 络 书 签

社 交 网 络

微 博

星 光 飘 移

实 时 时 间 天 气

图1 程序层次结构

Launcher通过调用LiveWallpaperPicker应用来实现对动态壁纸的调用和管理，通过AppWidgetManager来获取widget的信息，在Launcher内进行初始化后将其添加至页面编辑。锁屏与任务管理器作为单独的应用存在。

# 4 CSCI 部件设计

## 4.1 Launcher

### 4.1.1 功能描述

Launcher，即启动器，作为Android操作系统的用户界面，为用户使用Android系统提供接口。它的主要功能包括应用程序菜单、多桌面以及页面编辑。

### 4.1.2 部件组成

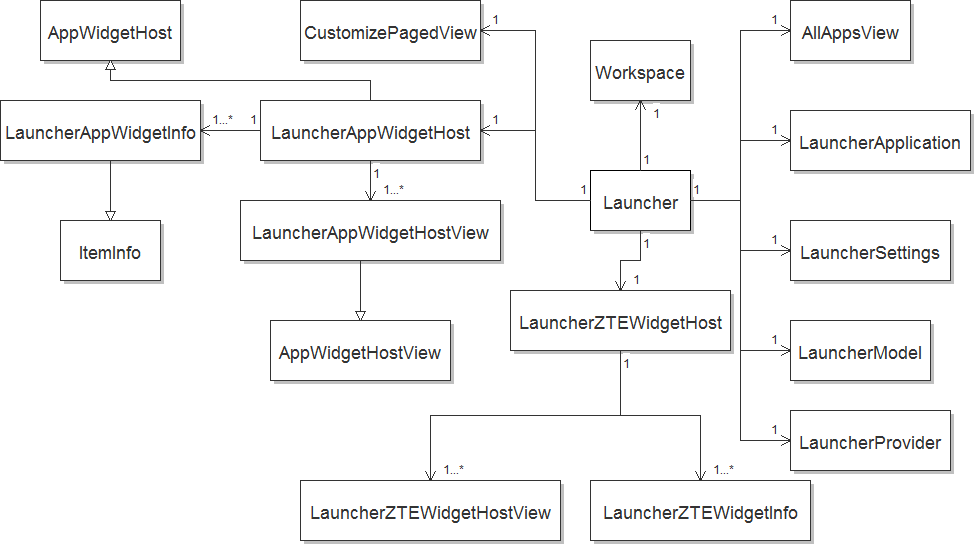


图2 Launcher架构

#### 4.1.2.1 多桌面

多桌面主要实现按钮栏以及五个水平排列的桌面。按钮栏上可以调用搜索应用、页面指示、调用应用程序菜单、调用页面编辑以及删除桌面widget。每个桌面都可以承载widget和应用程序快捷方式，进行桌面切换时，调用如需求所示的3D效果。

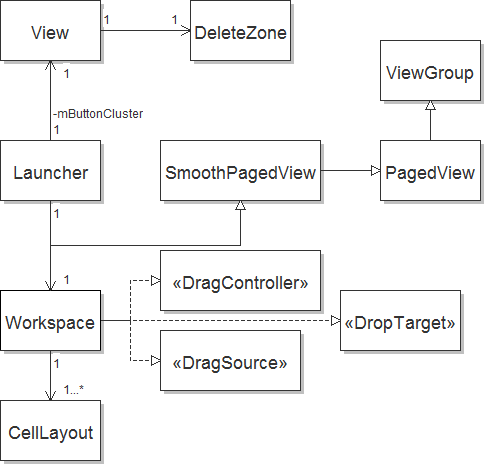


图3 多桌面架构

#### 4.1.2.2 应用程序菜单

应用程序菜单主要实现对应用程序的管理，包括以九宫格形式显示所有已安装的应用程序、启动应用程序、卸载应用程序、查看应用程序信息以及以拖拽形式在桌面创建应用程序的快捷方式。

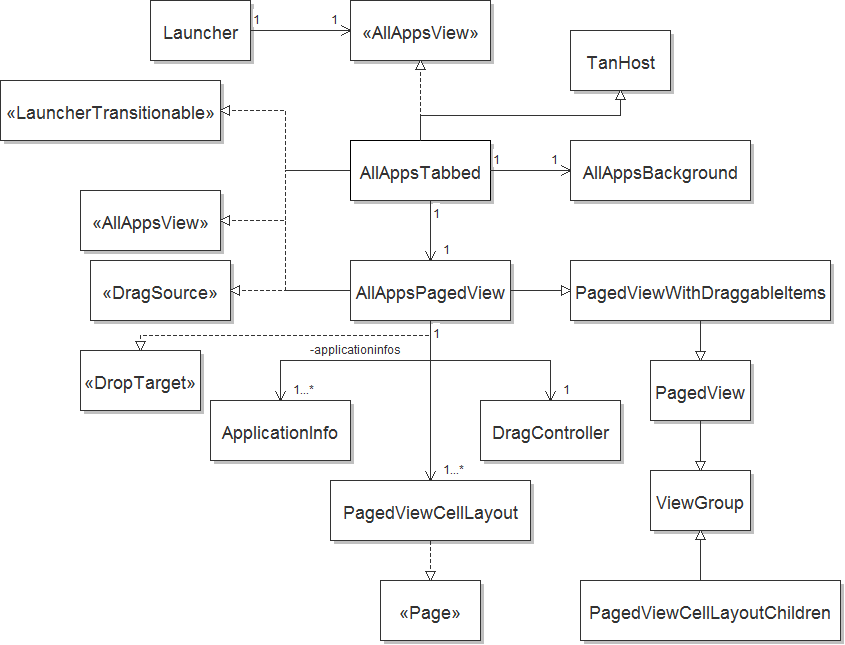


图4 应用程序菜单架构

#### 4.1.2.3 页面编辑

页面编辑主要实现AppWidget管理、应用程序快捷方式管理、壁纸管理。AppWidget管理通过AppWidgetManager获取当前系统已安装AppWidget的信息，并以水平列表的形式显示所有AppWidget的预览图，与Workspace进行交互，将AppWidget拖拽至桌面。应用程序快捷方式管理通过PackageManager获取当前系统已安装应用程序的图标，同样通过与Workspace进行交互，将快捷方式拖拽至桌面。壁纸管理通过调用LiveWallpaperPicker来实现对动态壁纸的管理，通过调用WallpaperChooser来实现对普通壁纸的管理。

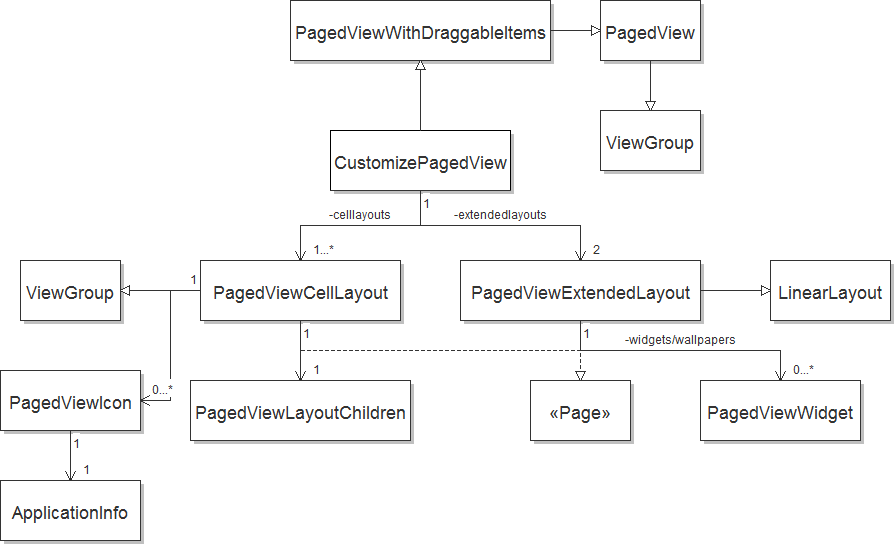


图5 页面编辑架构

### 4.1.3 接口设计

Launcher：Android启动器的主类，主要负责界面的初始化，数据库的读写，在桌面上加载widget，调用系统的接口，包括壁纸管理和应用程序管理等。

LauncherApplication：配置Launcher应用信息，包括加载数据库，注册intent接收器。

LauncherSettings：静态全局变量的定义。

LauncherModel：提供数据库读写的接口。

LauncherProvider：创建数据库，表favorite保存Launcher的配置信息。

LauncherZTEWidgetHost：提供创建ZTE自定义widget视图的接口。

LauncherZTEWidgetInfo：主要负责维护ZTE自定义widget的信息，并写入数据库。

LauncherZTEWidgetHostView：ZTE自定义widget的父类视图，并支持拖拽。

LauncherAppWidgetHost：提供创建系统原生widget视图的接口。

LauncherAppWidgetHostView：系统原生widget的视图，并支持拖拽。

LauncherAppWidgetInfo：负责维护系统原生widget的信息，并写入数据库。

ItemInfo：维护数据库列表项，并统一存入数据库。

Workspace：负责实现Launcher的多桌面，默认有5个桌面，并且可以水平方向上切换。

CellLayout：负责实现单个桌面，桌面上可以加载widget和应用程序快捷方式。

SmoothPagedView：实现桌面平滑地切换功能。

PagedView：负责多桌面的布局。

DragController：拖拽控制。

DragSource：Drag源接口。

DropTarget：Drop目标接口。

CustomizePagedView：实现页面编辑功能，负责与桌面进行交互，将widget、应用程序快捷方式拖入桌面。

PagedViewWithDraggableItems：支持拖拽的多页面布局，用于应用程序菜单中。

PagedViewCellLayout：负责实现应用程序菜单中的单页，以九宫格形式体现。

PagedViewCellLayoutChildren：应用程序菜单中单个图标布局。

PagedViewExtendedLayout：单行形式的多页面布局。

Page：页面接口，负责提供实现child的索引。

PagedViewWidget：单行形式的多页面布局。

PagedViewIcon：负责渲染页面中的单元。

AllAppsView：应用程序菜单视图。

AllAppsTabbed：Tab形式的应用程序菜单。

AllAppsBackground：应用程序菜单背景。

AllAppsPagedView：应用程序菜单的多页面布局实现。

## 4.2 动态壁纸

### 4.2.1 星光漂移

#### 4.2.1.1 功能描述

桌面动态壁纸功能，在锁屏状态时正常显示，在调出Menu菜单和Edit设置时作为背景置灰显示，动态效果如图6所示。

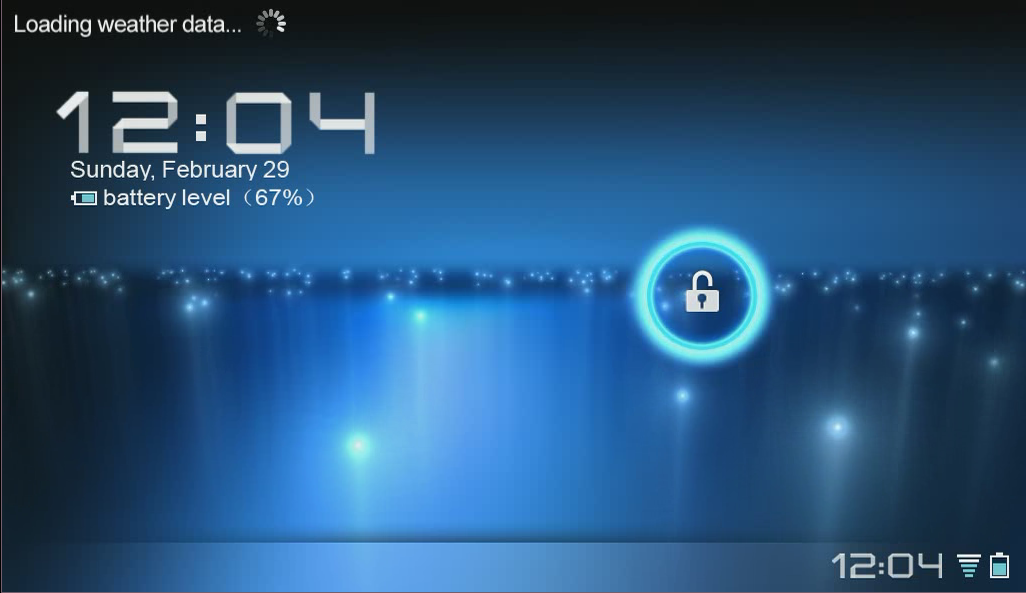


图6 星光漂移动态壁纸

#### 4.2.1.2 部件组成

**1. 部件一：动态壁纸服务实现**

构建一张动态壁纸必须继承实现系统定制的WallpapaerService，完成服务的部分函数实现，才可以被LivePicker动态壁纸加载器识别，进行正确加载。

**2. 部件二：动态壁纸服务引擎实现**

构建动态壁纸必须继承实现动态壁纸引擎，完成引擎提供的画布具体实现。

**3. 部件三：RenderScript C代码句柄**

创建RenderScript C代码对应的Java类，为Framework层调用提供接口。

**4. 部件四：加载图片线程**

创建线程，为RenderScript C渲染提供图片绑定。

**5. 部件五：RenderScript C代码**

完成最终的图片渲染功能。



图7 星光漂移动态壁纸部件组成

#### 4.2.1.3 接口设计

**1. 内部接口设计**

1）部件一与部件二

部件二为部件一的内部类，通过部件一的onCreateEngine（）方法返回部件二实例。

2）部件二与部件三

部件二中定义一类名为部件二的属性private StarWallpaperRS mRender，通过部件二提供的公有方法完成在部件二中SurfaceView中的画图工作。

3）部件三与部件四

部件四为部件三的内部类，在部件三中创建部件四的实例new LoadImageThread()。

4）部件三与部件五

由系统工具为两个部件构建完整的接口。

5）部件四与部件五

调用系统为部件三与部件五创建的接口，完成图片的加载工作。

**2. 外部接口设计**

1）部件一外部接口

部件一必须实现系统提供的WallpaperService。

2）部件二外部接口

部件二必须实现系统提供的WallpaperService.Engine。

### 4.2.2 实时时间天气

#### 4.2.2.1 功能描述

符合桌面动态壁纸功能，可以通过Edit设置进行管理，在锁屏时正常显示，在调出Menu菜单和Edit设置时作为背景置灰显示，动态效果如图8所示。



图8 实时时间天气动态壁纸效果

增加随时间和天气而实时改变的效果，系统默认存储 “晴天”、“雨天”两种天气，以及“清晨”（05:00～08:00）、“上午”（09:00～12:00）、“中午”（13:00～17:00）、“傍晚”（18:00～23:00）、“凌晨”（00:00～04:00）五个时段，启动时默认显示当前时段晴天的动态效果，每隔1分钟更新时段效果，每隔3分钟自动更换天气效果，如图9、10所示。



图9 晴天清晨的实时时间天气动态壁纸效果



图10 雨天傍晚的实时时间天气动态壁纸效果

#### 4.2.2.2 部件组成

**1. 部件一：动态壁纸服务实现**

构建一张动态壁纸必须继承实现系统定制的WallpaperService，完成服务的部分函数实现，才可以被LiveWallpaperPicker动态壁纸加载器识别，进行正确加载。

**2. 部件二：动态壁纸服务引擎实现**

构建动态壁纸必须继承实现动态壁纸引擎，完成引擎提供的画布具体实现。

**3. 部件三：RenderScript C代码句柄**

创建RenderScript C代码对应的Java类，为Framework层调用提供接口。

**4. 部件四：加载图片线程**

创建线程，为RenderScript C渲染提供图片绑定。

**5. 部件五：RenderScript C代码**

完成最终的图片渲染功能。

**5. 部件六：时段天气部件**

定时完成时段和天气的循环变换。

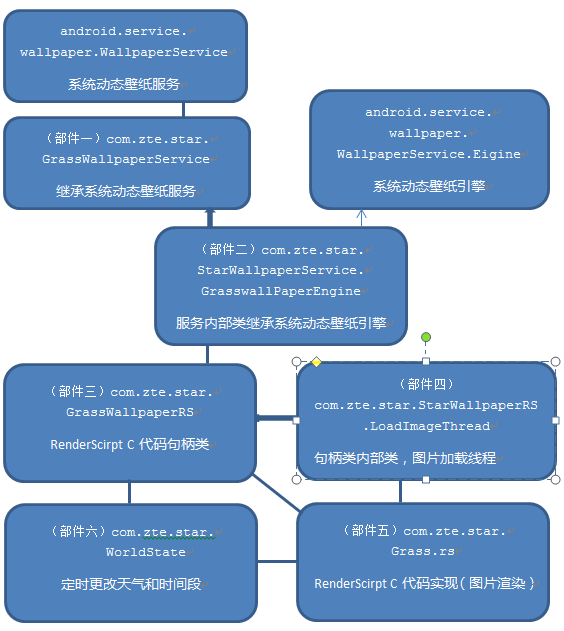


图11 实时天气动态壁纸部件组成

#### 4.2.2.3 接口设计

**1. 内部接口设计**

1）部件一与部件二

部件二为部件一的内部类，通过部件一的onCreateEngine（）方法返回部件二实例。

2）部件二与部件三

部件二中定义一类名为部件二的属性private StarWallpaperRS mRender，通过部件二提供的公有方法完成在部件二中SurfaceView中的画图工作。

3）部件三与部件四

部件四为部件三的内部类，在部件三中创建部件四的实例new LoadImageThread()。

4）部件三与部件五

由系统工具为两个部件构建完整的接口。

5）部件三与部件六

在部件三中创建部件六的实例WorldState = new WorldState()。

6）部件四与部件五

调用系统为部件三与部件五创建的接口，完成图片的加载工作。

7）部件五与部件六

调用系统为部件三与部件五创建的接口，完成天气和时段的变化工作。

**2. 外部接口设计**

1）部件一外部接口

部件一必须实现系统提供的WallpaperService。

2）部件二外部接口

部件二必须实现系统提供的WallpaperService.Engine。

## 4.3 桌面Widget

### 4.3.1 三维时间天气

#### 4.3.1.1 功能描述

实现桌面Widget的基本功能，可以被添加、删除、拖动等。

显示当前的时间、所在城市、天气信息。天气信息采用文字和动画效果结合的形式，根据不同的天气显示不同的动画效果。城市名称以文字形式显示在天气信息附近。当前时间采用三维立体数字形式显示，如图12所示。



图12 三维时间天气效果

系统默认存储“西安”、“北京”、“上海”三个城市、“晴天”、“雨天”两种天气，启动时默认显示当前时间西安晴天的信息，每隔1分钟采用翻转变化的动画效果更新时间，每隔3分钟自动更换天气效果。当天气信息为“雨天”时，雨滴能与星光漂移动态壁纸进行交互，水滴滴落的地方出现涟漪；雨滴滴落到三维立体时间数字上时，雨滴会落到数字上，并从数字上滴落，并与壁纸进行同样交互，如图13所示。



图13 下雨时三维时间天气交互效果

当用户在Widget中上下滑动时，可以切换城市，切换时三维立体数字与滑动方向旋转一个角度后还原，依次切换三个城市，并更换天气信息。

#### 4.3.1.2 部件组成

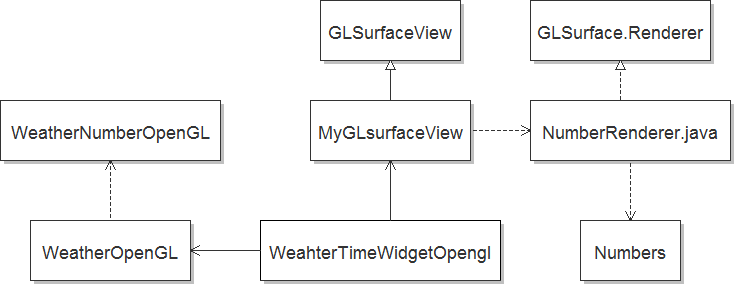


图14 时间天气组成部件

#### 4.3.1.3 接口设计

MyGLsurfaceView：加载显示时间的Renderer，并对相应的事件进行监控和处理。

NumberRenderer：绘制三维时间，三维数字的渲染和动画控制。

Numbers：提供三维数字模型的建立。

WeatherNumberOpenGL：对于资源的重定位，通过统一的方法来获得资源的resourceID。

WeatherOpenGL：天气的显示以及动画效果的处理。

WeatherTimeWidgetOpenGL：提供getView()方法，将整个三维天气的View返回。

### 4.3.2 网络书签

#### 4.3.2.1 功能描述

实现桌面Widget的基本功能，可以被添加、删除、拖动等。

以两排并列形式显示网络书签的缩略图，系统存储29个书签的缩略图，一屏长度可显示3张半缩略图，如图15所示。



图15 书签滑动查看效果

当用户在Widget中上下滑动时，所有书签缩略图滑动显示，并绕Y轴做一定角度的旋转，点击一张书签缩略图，会有震颤效果，且被点击的页面会从书签中飘出，并逐渐放大到整个屏幕，随后调用浏览器应用如图16所示。



图16 书签点击效果

#### 4.3.2.2 部件组成

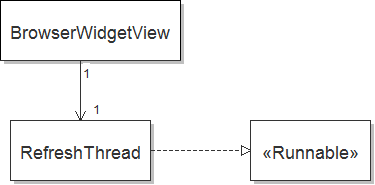


图17 网络书签组成部件

#### 4.3.2.3 接口设计

BrowserWidgetView：对数据进行初始化，对于事件的监听以及动画效果的显示和控制，提供getView()方法，将网络书签的View返回。

### 4.3.3 联系人

#### 4.3.3.1 功能描述

实现桌面Widget的基本功能，可以被添加、删除、拖动等。

以两排并列形式显示联系人的缩略图，系统存储联系人的缩略图，一屏长度可显示2张半缩略图，如图18所示。



图18 联系人效果

当用户在Widget中上下滑动时，所有书签缩略图滑动显示，并绕Y轴做一定角度的旋转，如图19所示。



图19 联系人滑动效果

#### 4.3.3.2 部件组成



图20 联系人组成部件

#### 4.3.3.3 接口设计

FavoriteWidgetView：对数据进行初始化，对于事件的监听以及动画效果的显示和控制，提供getView()方法，将联系人的View当做返回值返回。

### 4.3.4 社交网络

#### 4.3.4.1 功能描述

实现桌面Widget的基本功能，可以被添加、删除、拖动等。

显示好友的缩略页面，系统存储5个好友的缩略页面，当用户在Widget中向上滑动当前页面时，当前页以折纸方式向上退出，下个好友缩略页面由下往上滑入，向下滑动效果类似，如图21所示。



图21 社交网络翻页动画效果

#### 4.3.4.2 部件组成

1. 部件一：网络书签widget的启动Activity

负责一些布局的基础设置以及其他部件的一些初始化工作。

1. 部件二：网络书签widget的视图部件

通过部件三和部件四的支持生成widget的视图并呈现。对于滑动和点击事件的监听及响应的处理。实现点击和滑动事件时的动画效果。

1. 部件三：网络书签widget的view中的适配器部件ImageAdapter

视图中显示内容的适配器，用来初始化资源，向视图中添加要显示内容的资源。

1. 部件四：网络书签widget的view类型定义

用来定义显示view的类型。

#### 4.3.4.3 接口设计

1. 部件一与部件二：部件一通过new HoloView()方法实例部件二，并将其设置为当前显示。
2. 部件二与部件三：部件二通过new ImageAdapter()方法实例部件三。
3. 部件二与部件四：部件二通过findViewById获取部件四。

### 4.3.5 微博

#### 4.3.5.1 功能描述

实现桌面Widget的基本功能，可以被添加、删除、拖动等。

默认显示微博的界面，系统存储8条微博信息，一屏显示3条半微博信息，如图22所示。当用户在Widget中上下滑动时，所有微博信息滑动显示。



图22 微博默认界面

点击刷新按钮后为从上往下翻转滑入一条新微博。如图23所示。



图23 微博刷新动画效果

#### 4.3.5.2 部件组成



图24 微博组成部件

#### 4.3.5.3 接口设计

MicroblogWidgetView：事件的监听以及动画效果的实现，提供getView()方法将微博的视图当做一个View返回。

FixedListLayout：对于数据的初始化，以及视图布局的控制。

## 4.4 锁屏

### 4.4.1 功能描述

用户关闭屏幕时启动锁屏界面，再次开启屏幕时，显示锁屏界面。锁屏界面左侧显示时间、日期和电池状态。锁屏界面右侧为解锁按钮，长按按钮解锁，如图25所示。



图25 锁屏界面

### 4.4.2 部件组成

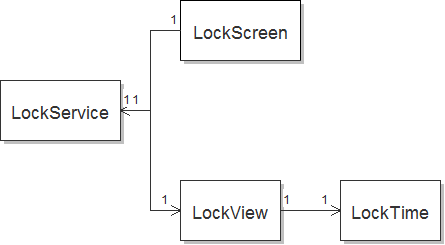


图26 锁屏部件组成

### 4.4.3 接口设计

LockScreen：主要负责实现开启/停止锁屏服务，禁用键盘；

LockService：锁屏服务，注册广播接收器，监控屏幕关闭操作，一旦屏幕关闭，启动锁屏界面；

LockView：锁屏界面，长按解锁；

LockTime：锁屏界面时的时间显示。

## 4.5 任务管理器

### 4.5.1 功能描述

任务管理器启动时，以弧形方式显示当前正在运行的程序的缩略图，从屏幕左边飞入，如图27所示。



图27 多任务程序缩略图效果

用户可上下滑动查看所有的程序缩略图，滑动时按照弧形方向转动，程序缩略图有前后震动效果，关闭程序时，上下两个临近程序缩略图会有上下震动效果。

点击一个程序缩略图，被点击的缩略图会飘出，并逐渐放大到整个屏幕，然后激活该应用程序，如图28所示。



图28 程序缩略图点击效果

### 4.5.2 部件组成

1. 部件一：任务管理器的扇形任务列表

当启动任务管理器的时候，扇形任务列表装载任务信息，从屏幕左侧飞入。

1. 部件二：扇形任务列表的拨动效果

监听用户手指的纵向动作，将捕获的手指位移由直角坐标系转化为极坐标系的角度值，控制扇形任务列表的旋转，并做减速和回转动画效果。并根据数值大小使用高速转动和低速转动两种速度调整模式。

1. 部件三：扇形任务列表的删除效果

用户点击任务列表中的红叉区域，指定的任务选项消失，两侧任务选项合并，完成任务选项的删除，同时关闭对应的任务进程。

1. 部件四：扇形任务列表的点选效果

监听用户手指动作，当认定为点选时，指定任务选项震动，伴随飞出效果，开启相应应用程序。

### 4.5.3 接口设计

1. 部件一与部件二：部件一通过drawPan(GL10)方法实例部件二。
2. 部件一与部件三：部件一通过removeItem(int)方法实例部件三。
3. 部件一与部件四：部件一通过pressItem(GL10, int)获取部件四。