

# bigapple之加载网络图片

## 1、问题需求

有一种需求是这样的，需要从网络上加载图片。例如，加载头像，加载广告图片等。如果从头自己写，需要考虑很多问题，例如：

- (1) 缓存问题，你不可能每次都去网络加载，用户那点可怜流量被你这么白白浪费可不砍死你。
- (2) 说到缓存，你用什么缓存呢。据说一个应用安卓只分配给你16M内存，有些是24M根据不同手机而定。而现在的一张图片就十几M，两个图片就几十M，这就不行了。
- (3) 在加载多张图片快速切换时，一般的会出现卡顿现象，这就需要特殊处理。

## 2、解决方案

俗话说的好，有问题有需求，就会有解决方法。这不，bigapple1.1.2之后加入了一个强大的加载网络图片。你完全不用考虑上面种种问题，使用他的基本原理是同时使用了两种缓存。内存缓存和磁盘缓存。每次都会先从内存缓存中取，没有再从磁盘中，最后才去网络地址上加载。当然你不过建议是两个同时使用。内存缓存快但容量小，一般只会分配个8M都嫌多了。磁盘缓存相对多点，但是访问比内存慢，但是比起网络加载，那

- (1) 默认最简单的使用例子

```
public static void loadBitmapDefault(Context context, ImageView imageView) {
    AnBitmapUtils anBitmapUtils = AnBitmapUtils.create(context);
    anBitmapUtils.display(imageView, "http://img7.9158.com/200709/01/11/53/200709018758949.jpg");
}
```

上面的context你懂的，不懂那就先别做安卓了。imageView就是加载图片的控件对象。下面写死的就是加载图片的地址，是个美女图哦。是不是

- (2) 自定义一些参数使用例子

```
public static void loadBitmapCustom(Context context, ImageView imageView) {
    AnBitmapUtils anBitmapUtils = AnBitmapUtils.create(context, ContextUtils.getSdCardPath() +
        "/bigapple_test/");
    Bitmap loadingBitmap = BitmapFactory.decodeResource(context.getResources(),
        R.drawable.loading);
    Bitmap failBitmap = BitmapFactory.decodeResource(context.getResources(), R.drawable.fail);
    anBitmapUtils.display(imageView, "http://img7.9158.com/200709/01/11/53/200709018758949.jpg",
        loadingBitmap, failBitmap);
}
```

loadingBitmap是你在加载中显示的图片。failBitmap是你在加载失败之后显示的图片。这两种情况无论是写web还是什么的都很常见的吧。还有 + "/bigapple\_test/" 这个是一个磁盘缓存的存放路径。

这个可以你自己定义的。当然你也可以设置成不要磁盘缓存。不过最好要，内存缓存那么点，只使用内存缓存，两张大图下来没什么效果。

- (3) 有缓存么，当然要刷一下。

```
public static void clearCache(Context context) {
    AnBitmapUtils anBitmapUtils = AnBitmapUtils.create(context);
    anBitmapUtils.clearCache();
}
```

clearCache这个方法就刷新所有缓存，包括缓存在内存中和磁盘中的。当然还是你可以指定刷某一个缓存，你试一下就知道。是不是有人要问，放心这个内部是单例的。所以你在任何地方create出来的都是同一个对象。

## 3、写在最后

当然，一个人的力量有限。copy代码不可耻。这部分代码参考，网络上比较流行的安卓开源框架afinal和xutils。如有雷同纯属巧合哈。