Picasso：一个专为Android打造的强大的图片下载和缓存库

## 简介

在Android应用中，图片消费了大量的资源，却为应用提供了很好的视觉体验。幸运的是，Picasso为你的应用提供了非常容易的图片加载方式——通常一行代码就可以搞定！

**Picasso.with(context).load("http://i.imgur.com/DvpvklR.png").into(imageView);**

Picasso处理了Android开发中图片加载的许多坑：

* 在Adapter中，处理了ImageView的循环利用和取消下载。
* 耗费最小的内存处理了复杂的图形变化。
* 在内存和磁盘上自动缓存图片。



## 特点

### Adapter中的下载

自动检测adapter中的重用功能，且一旦发现重用，将自动取消之前的下载。

**@Override public void getView(int position, View convertView, ViewGroup parent) {**

**SquaredImageView view = (SquaredImageView) convertView;**

**if (view == null) {**

**view = new SquaredImageView(context);**

**}**

**String url = getItem(position);**

**Picasso.with(context).load(url).into(view);**

**}**

### 图片变换

变换图片以便更好的适应布局，同时也减少了内存的使用。

**Picasso.with(context)**

**.load(url)**

**.resize(50, 50)**

**.centerCrop()**

**.into(imageView)**

你也可以自定义图片的转换方式以达到更高的变换要求。

**public class CropSquareTransformation implements Transformation {**

**@Override public Bitmap transform(Bitmap source) {**

**int size = Math.min(source.getWidth(), source.getHeight());**

**int x = (source.getWidth() - size) / 2;**

**int y = (source.getHeight() - size) / 2;**

**Bitmap result = Bitmap.createBitmap(source, x, y, size, size);**

**if (result != source) {**

**source.recycle();**

**}**

**return result;**

**}**

**@Override public String key() { return "square()"; }**

**}**

将**CropSquareTransformation**这个类的一个实例传递给transform()函数即可。

### 占位图片

Picasso同时支持“正在下载”时和“图片下载出错”后这两种状态展示的默认图片。

**Picasso.with(context)**

**.load(url)**

**.placeholder(R.drawable.user\_placeholder)**

**.error(R.drawable.user\_placeholder\_error)**

**.into(imageView);**

Picasso至多会尝试三次下载，如果三次下载都失败了，就会在原图位置上展示“图片下载出错”时的图片。

### 资源加载

Picasso支持resources、assets、files和content provider作为图片的加载源。

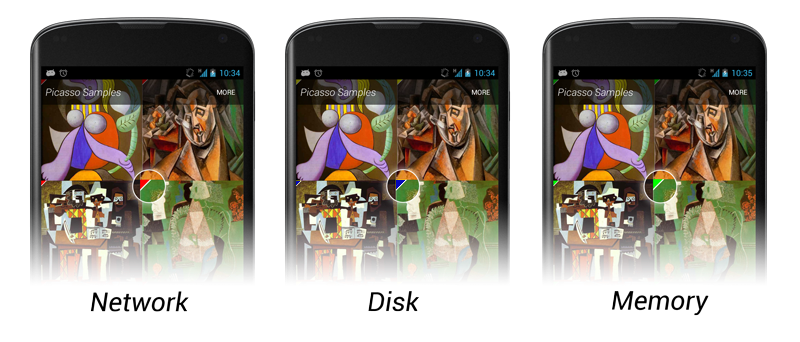
**Picasso.with(context).load(R.drawable.landing\_screen).into(imageView1);**

**Picasso.with(context).load("file:///android\_asset/DvpvklR.png").into(imageView2);**

**Picasso.with(context).load(new File(...)).into(imageView3);**

### 调试指示器

作为开发者，你可以在图片的左上角展示一个彩色的小三角，不同的颜色指明了图片资源的不同来源。调用Picasso对象的setIndicatorsEnabled(true)方法就可以了。



## 下载

<http://repo1.maven.org/maven2/com/squareup/picasso/picasso/2.5.2/picasso-2.5.2.jar>

Picasso的源代码，例子和这个网址（http://square.github.io/picasso/）都放在了[GitHub](http://github.com/square/picasso)上

**MAVEN**

<dependency>

<groupId>com.squareup.picasso</groupId>

<artifactId>picasso</artifactId>

<version>2.5.2</version>

</dependency>

**GRADLE**

compile 'com.squareup.picasso:picasso:2.5.2'

## 为Picasso做贡献

如果你想为Picasso开发贡献你的代码，你可以上github，然后fork该代码库，再把你的修改发给我们（pull request）。

在提交代码的时候，请保持代码的习惯和风格与原来的一致，以便于尽可能保持代码的可阅读性。同时，为了保证你的代码正确编译，请运行mvn clean verify。

此外，你需要同意[Individual Contributor License Agreement (CLA)](http://squ.re/sign-the-cla)才能将你的代码合并到Picasso工程中。

## 重要资料

[1]Javadoc 相关api介绍: <http://square.github.io/picasso/javadoc/index.html>

[2]StackOverflow 相关问题 :<http://stackoverflow.com/questions/tagged/picasso?sort=active>

## 许可证

Copyright 2013 Square, Inc.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

原文链接：<http://square.github.io/picasso/>