前webapp越来越多，体验也越来越好，为了能够更好的使用WebView展示出流畅的的页面，可以从以下几点做优化：

* **WebView缓存**
* **资源文件本地存储**
* **减少耗时操作**
* **客户端UI优化**

WebView缓存

开启WebView的缓存功能可以减少对服务器资源的请求，一般使用默认缓存策略就可以了。

*//设置 缓存模式* webView.getSettings().setCacheMode(WebSettings.LOAD\_DEFAULT); *// 开启 DOM storage API 功能* webView.getSettings().setDomStorageEnabled(**true**);

### 资源文件本地存储

资源等文件(不需要更新)本地存储，在需要的时候直接从本地获取。哪些资源需要我们去存储在本地呢，当然是一些不会被更新的资源，例如图片文件，js文件，css文件，替换的方法也很简单，重写WebView的方法即可。

{

try {

if (url.endsWith("icon.png")) {

InputStream is = appRm.getInputStream(R.drawable.icon);

WebResourceResponse response = new WebResourceResponse("image/png",

"utf-8", is);

return response;

} else if (url.endsWith("jquery.min.js")) {

InputStream is = appRm.getInputStream(R.raw.jquery\_min\_js);

WebResourceResponse response = new WebResourceResponse("text/javascript",

"utf-8", is);

return response;

}

} catch (IOException e) {

e.printStackTrace();

}

return super.shouldInterceptRequest(view, url);

}

1. *appRm为app资源管理器，读取drawable，assets，raw下的资源，都是android系统的一些很简单的函数调用。*
2. *getInputStream的参数代表资源具体位置*
3. *WebResourceResponse后的资源类型需要写正确*

### 客户端UI优化

怎么让用户看不到WebView加载前的白色页面呢？首次加载后页面的跳转可以用上面的步骤进行优化，可以提供给用户一个很好的体验，那加载的第一页呢？我们需要WebView预加载页面，这个该怎么做到的呢？下面提供两种方法:

* ViewPager，将欢迎页面与WebView页面一起放进ViewPager中，设置预加载页面个数，使WebView所在页面可以预加载，在加载完毕的时候切换到WebView所在页面。
* FrameLayout，将欢迎页面与WebView页面的布局合在一起，显示在一个页面内，起始隐藏WebView布局，待WebView加载完毕，隐藏欢迎布局，显示WebView布局。

使用FrameLayout简单一些，两种方法都是需要对WebChromeClient的onProgressChanged进行监听，加载完毕进行页面切换，如下：