**合辑简介优化简介**

本次的业务文档就说明一下客户端里合辑简介优化的内容吧。

首先，本次合辑简介优化最直观的感受，就是，在进入合辑简介页面时，会看到骨架屏（在页面内容未加载完成之前，先使用一些图形进行占位，待内容加载完成之后再它替换掉）的展示。

其次，就是该次优化，引用了不同的缓存方案，对不同环境下进行兼容，使得在使用各种手机进入该简介页面时，内部处理效果不同，直观也会略有区别。不急，待我慢慢说来。

那好，我们下面就简单的开始介绍本次合辑简介优化都有哪些改动吧。

第一个改动，本次的优化由以前的单页应用改为现在的多页应用，当然，这些直观你是看不出来的，我们可以使用charles抓包，当你第一次进入合辑简介时，可以抓包看到，html请求有多个。

第二个改动，本次的优化引入的缓存方案，可以在不同情况下，使用不同的缓存方案：

1.当你进入合辑简介页面时会加载一个.appcache文件，该文件有点好处就是，可以离线使用，当你的手机没有网络时，再进入合辑简介页面，就会展示出我们预先在.appcache文件里缓存的内容。如果你是第一次进入合辑简介页面，且处于没有网络的状态，那么，当我没有说。

2.缓存资源的分开部署。我们将html和js/css/img分开部署，在我们不进行内容更新时，用户在进入合辑简介页面时，不会再次请求这几个文件，这样，看起来，是不是进入合辑简介页面的时候，稍微快了那么一点点呢...

3.js的异步执行。JavaScript语言的执行环境是“单线程”，所以，当有一个任务加载耗时时间较长，后面的任务都必须排队等着，万一哪天抽风这个代码万一死循环了，给到我们的感受就是，死慢，非常慢，我的天怎么这么慢，后面的任务还得眼巴巴的等着，一边诅咒前面的进程，一边跃跃欲试的等着被执行。所以，本次优化，就做成了异步执行。这样，万一哪天有个任务抽风挂掉了，后面的任务可以换个队伍继续被执行。虽然你会看到这个页面有些东西没有显示出来，但是，也总比你什么都看不到要强那么一丢丢的吧。

好的，那我们和以前的加载方式，来一次对比吧。

直观感受加载速度：

旧版：1.首先进入合辑简介页面，一个进度条使劲的加载，页面是空白的，等到所有请求全部完成后，页面内容展示出来了，这个时候，你已经喝了一杯咖啡；2.万一网络有些故障，速度不是那么的快，你进入合辑简介页面，对，没错，还是一个空白页面，一个进度条拼了命的加载，你望眼欲穿的等着，当然，你也可以和上面一样，喝杯咖啡，在顺便泡个澡。

新版：1.首先进入合辑简介页面，一切都是那么有条不紊的进行，在你拿起咖啡杯的时候，页面已经出来了；2.万一网络有些故障，速度不是那么的快，你进入合辑简介页面，不说其他，好歹你可以看到一个骨架，知道这个页面大致的样子，其余内容，你可以脑补啊。

虽然上方内容有些夸大的成分，但是，本次页面的优化相对于原来的样子来说已经好了很多。

所以，本次介绍完~