

转

变态的静态资源缓存与更新（超详细好文）

2016年04月16日 10:49:32

zhangjs712

阅读数：9630

标签：

浏览器

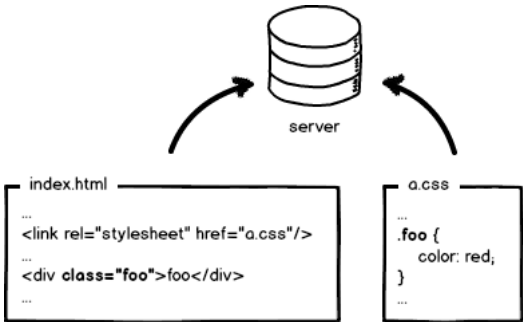
前端开发

更多

这是一个非常有趣的 非主流前端领域，这个领域要探索的是如何用工程手段解决前端开发和部署优化的综合问题，入行到现在一直在学习和实践中。

在我的印象中，facebook是这个领域的鼻祖，有兴趣、有梯子的同学可以去看看facebook的页面源代码，体会一下什么叫工程化。

接下来，我想从原理展开讲述，多图，较长，希望能有耐心看完。



让我们返璞归真，从原始的前端开发讲起。上图是一个“可爱”的index.html页面和它的样式文件a.css，用文本编辑器写代码，无需编译，本地预览，确认OK，丢到服务器，等待用户访问单，好好玩啊，门槛好低啊，分分钟学会木有！

Name Path	Status Text	Type	Size	Timeline
index.html	200	text/html	1.0KB	<div></div>
a.css	200	text/css	1.0KB	<div></div>

然后我们访问页面，看到效果，再查看一下网络请求，200！不错，太™完美了！那么，研发完成。。。了么？

等等，这还没完呢！对于大公司来说，那些变态的访问量和性能指标，将会让前端一点也不“好玩”。

看看那个a.css的请求吧，如果每次用户访问页面都要加载，是不是很影响性能，很浪费带宽啊，我们希望最好这样：

Name Path	Status Text	Type	Size	Timeline
index.html	200	text/html	1.0KB	<div></div>
a.css	304	text/css	280B	<div></div>

利用304，让浏览器使用本地缓存。但，这样也够了吗？不成！304叫协商缓存，这玩意还是要和服务器通信一次，我们的优化级别是变态级，所以必须彻底灭掉这个请求，变成这样：

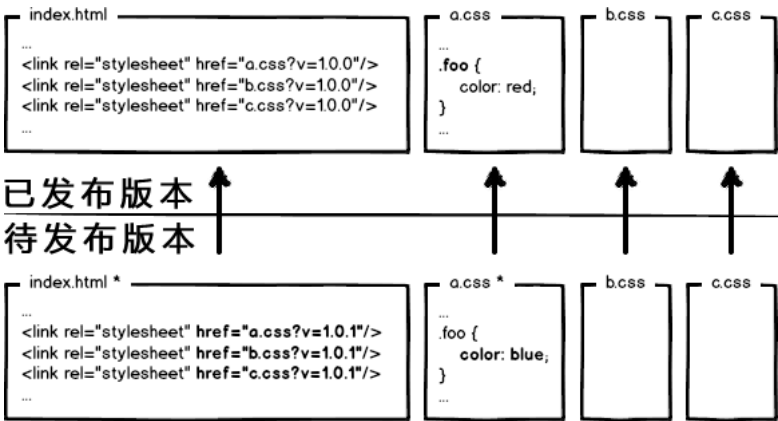
Name Path	Status Text	Type	Size	Timeline
index.html	200	text/html	1.0KB	<div></div>
a.css	200	text/css	(from cache)	<div>I</div>

强制浏览器使用本地缓存（cache-control/expires），不要和服务器通信。好了，请求方面的优化已经达到变态级别，那问题来了：你都不让浏览器发资源请求了，这缓存咋更新？

index.html

```
...
<link rel="stylesheet" href="a.css?v=1.0.0"/>
...
<div class="foo">foo</div>
...
```

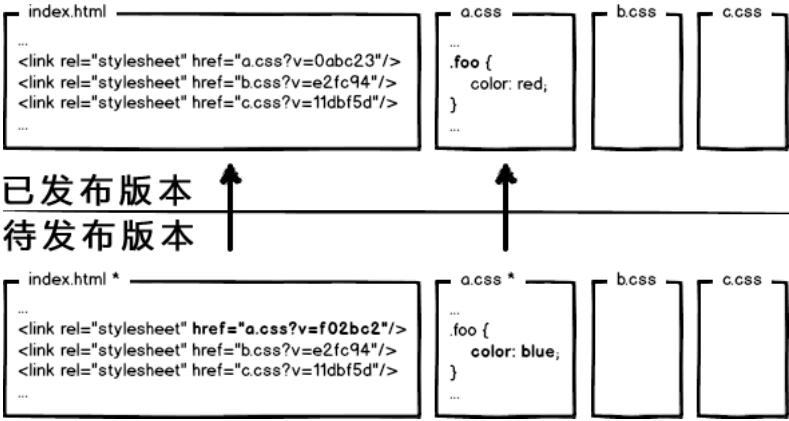
下次上线，把链接地址改成新的版本，就更新资源了不是。OK，问题解决了么？！当然没有！大公司的变态又来了，思考这种情况：



页面引用了3个css，而某次上线只改了其中的a.css，如果所有链接都更新版本，就会导致b.css，c.css的缓存也失效，那岂不是又有浪费了？！

重新开启变态模式，我们不难发现，要解决这种问题，必须让url的修改与文件内容关联，也就是说，只有文件内容变化，才会导致相应url的变更，从而实现文件级别的精确缓存控制。

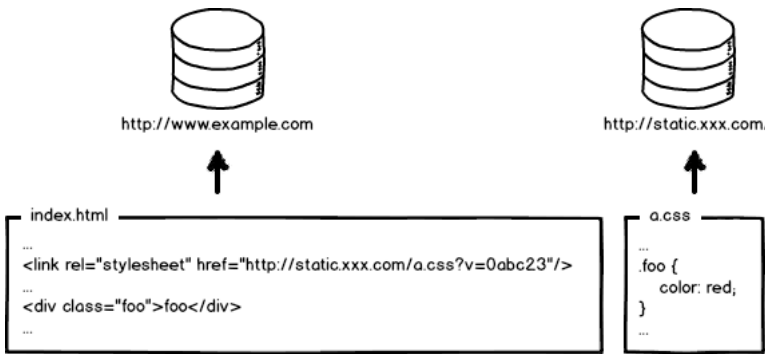
什么东西与文件内容相关呢？我们会很自然的联想到利用 数据摘要要算法 对文件求摘要信息，摘要信息与文件内容——对应，就有了一种可以精确到单个文件粒度的缓存控制依据了。好了，主要信息的：



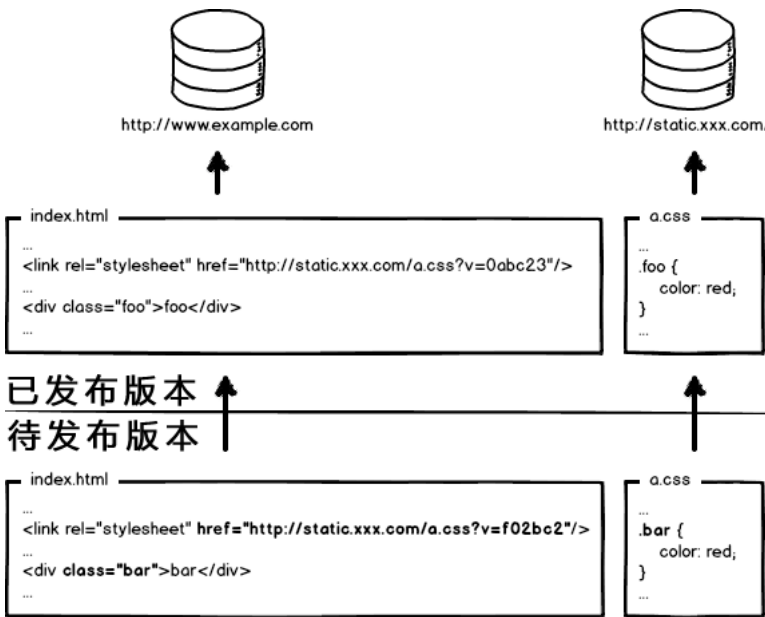
这回再有文件修改，就只更新那个文件对应的url了，想到这里貌似很完美了。你觉得这就够了么？大公司告诉你：图样图森破！

唉~~~~，让我喘口气

现代互联网企业，为了进一步提升网站性能，会把静态资源和动态网页分集群部署，静态资源会被部署到CDN节点上，网页中引用的资源也会变成对应的部署路径：



好了，当我要更新静态资源的时候，同时也会更新html中的引用吧，就好像这样：



这次发布，同时改了页面结构和样式，也更新了静态资源对应的url地址，现在要发布代码上线，亲爱的前端研发同学，你来告诉我，咱们是先上线页面，还是先上线静态资源？

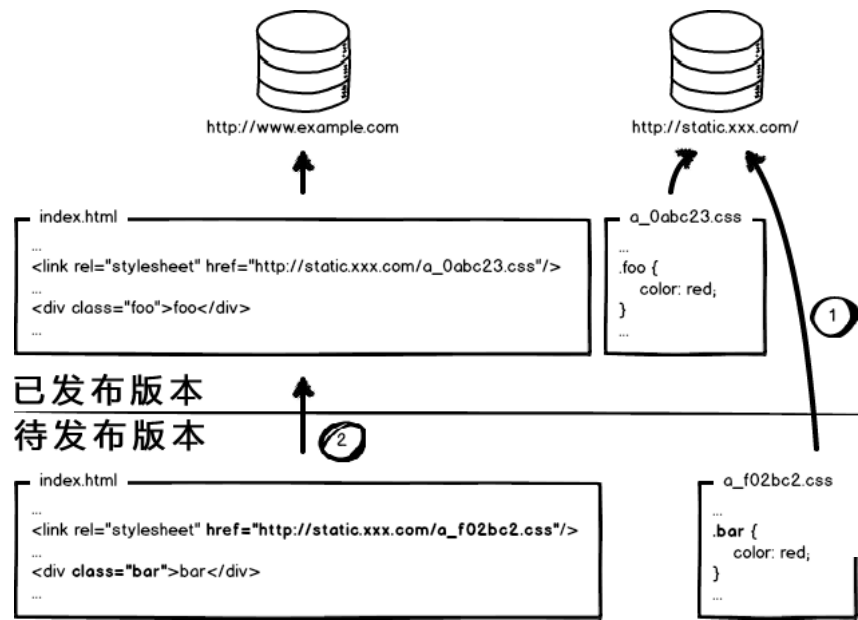
先部署页面，再部署资源：在二者部署的时间间隔内，如果有用户访问页面，就会在新的页面结构中加载旧的资源，并且把这个旧版本的资源当做新版本缓存起来，其结果就是：用户访问到错乱的页面，除非手动刷新，否则在资源缓存过期之前，页面会一直执行错误。

先部署资源，再部署页面：在部署时间间隔之内，有旧版本资源本地缓存的用户访问网站，由于请求的页面是旧版本的，资源引用没有改变，浏览器将直接使用本地缓存，这种情况下页面没有本地缓存或者缓存过期的用户访问网站，就会出现旧版本页面加载新版本资源的情况，导致页面执行错误，但当页面完成部署，这部分用户再次访问页面又会恢复正常了。

好的，上面一坨分析想说的就是：先部署谁都不成！都会导致部署过程中发生页面错乱的问题。所以，访问量不大的项目，可以让研发同学苦逼一把，等到半夜偷偷上线，先上静态资源，再部署页面少一些。

但是，大公司超变态，没有这样的“绝对低峰期”，只有“相对低峰期”。So，为了稳定的服务，还得继续追求极致啊！

这个奇葩问题，起源于资源的覆盖式发布，用待发布资源覆盖已发布资源，就有这种问题。解决它也好办，就是实现非覆盖式发布。

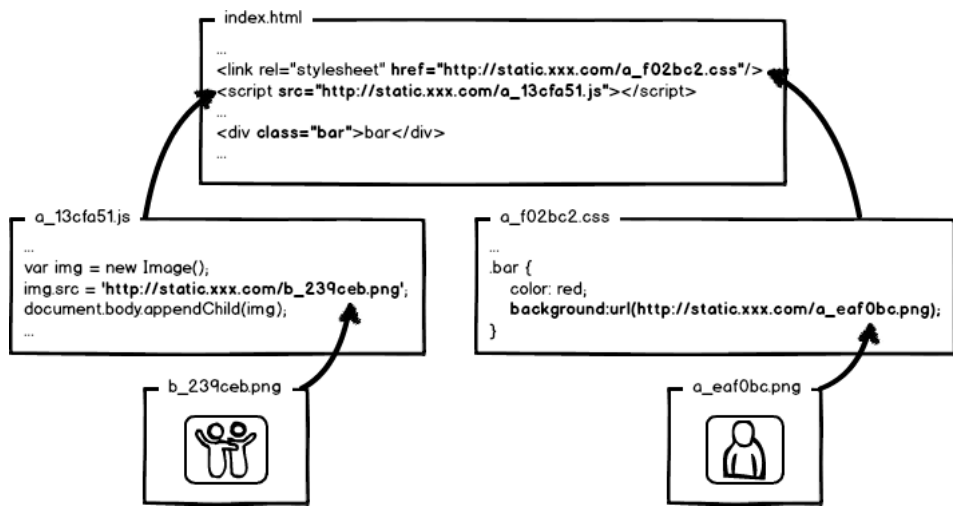


看上图，用文件的摘要信息来对资源文件进行重命名，把摘要信息放到资源文件发布路径中，这样，内容有修改的资源就变成了一个新的文件发布到线上，不会覆盖已有的资源文件。上线过程态资源，再灰度部署页面，整个问题就比较完美的解决了。

所以，大公司的静态资源优化方案，基本上要实现这么几个东西：

- 1.配置超长时间的本地缓存 —— 节省带宽，提高性能
- 2.采用内容摘要作为缓存更新依据 —— 精确的缓存控制
- 3.静态资源CDN部署 —— 优化网络请求
- 4.更资源发布路径实现非覆盖式发布 —— 平滑升级

全套做下来，就是相对比较完整的静态资源缓存控制方案了，而且，还要注意的，静态资源的缓存控制要求在 前端所有静态资源加载的位置都要做这样的处理。是的，所有！什么js、css自s、css文件中引用的资源路径，由于涉及到摘要信息，引用资源的摘要信息也会引起引用文件本身的内容改变，从而形成级联的摘要变化，大概示意图就是：



大数据 开发月薪35K+！可是，多数程序员会学吗？

大数据技术与运用的成熟，应用集中于互联网、金融、医疗、新能源、通信和房地产等行业。整理平均薪资情况和大数据学习大纲供查看

想对作者说点什么？

我来说两句

tousss： 好久了，不知道博主还记的这个摘要信息怎么获得？ c或java处理后返回么？ (08-11 09:23 #4楼)

前端老王： 深度好文 (06-02 10:47 #3楼)