二次元与CDN 技术联姻的正确打开方式

吴学军

2017 年 8 月 12 日

又拍云

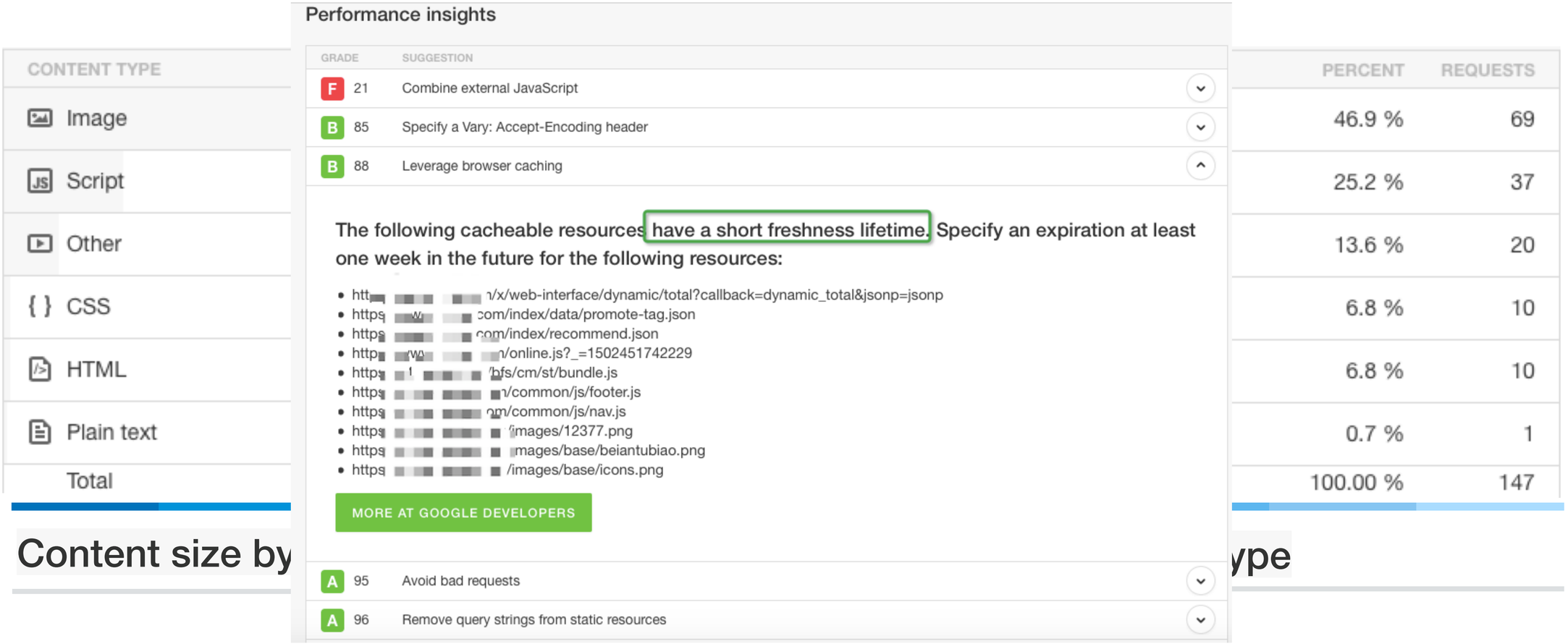
分享大纲

- PART1：前言
- PART2：通用优化篇
- PART3：图片、视频篇
- PART4：HTTPS 篇

PART
01

前言

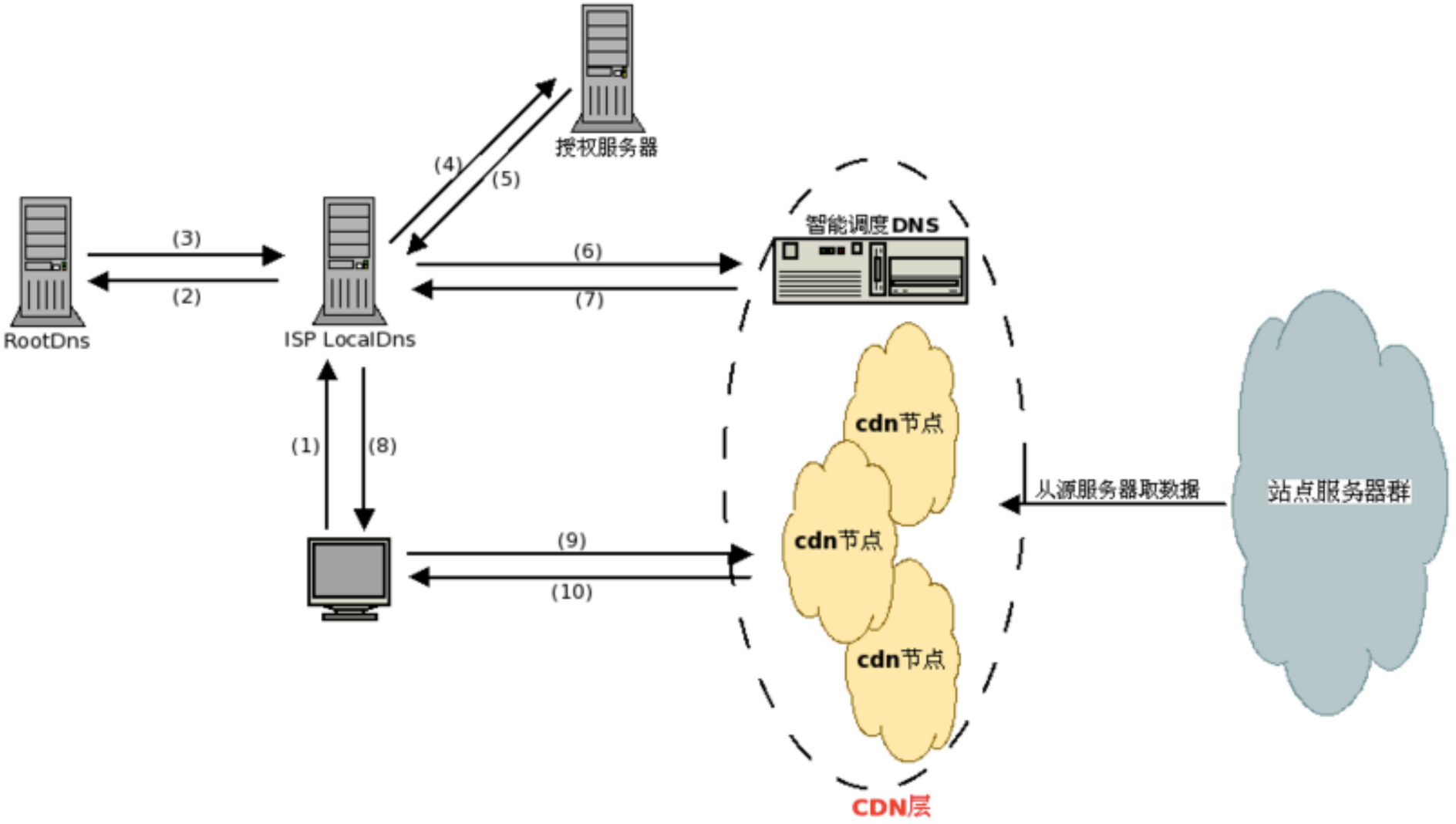
二次元网站网站元素分析



图片以及视频占到 70% 以上，其他主要是文本、HTML、JS、CSS 等

二次元网站遇到的挑战

CDN 整体架构图



CDN 对网络的优化作用

- 解决服务端的第一公里问题
- 缓解甚至消除了不同运营商之间互联网的瓶颈造成的影响
- 减轻了各省的出口带宽压力
- 缓解了骨干网的压力
- 优化了网上热点内容的分布

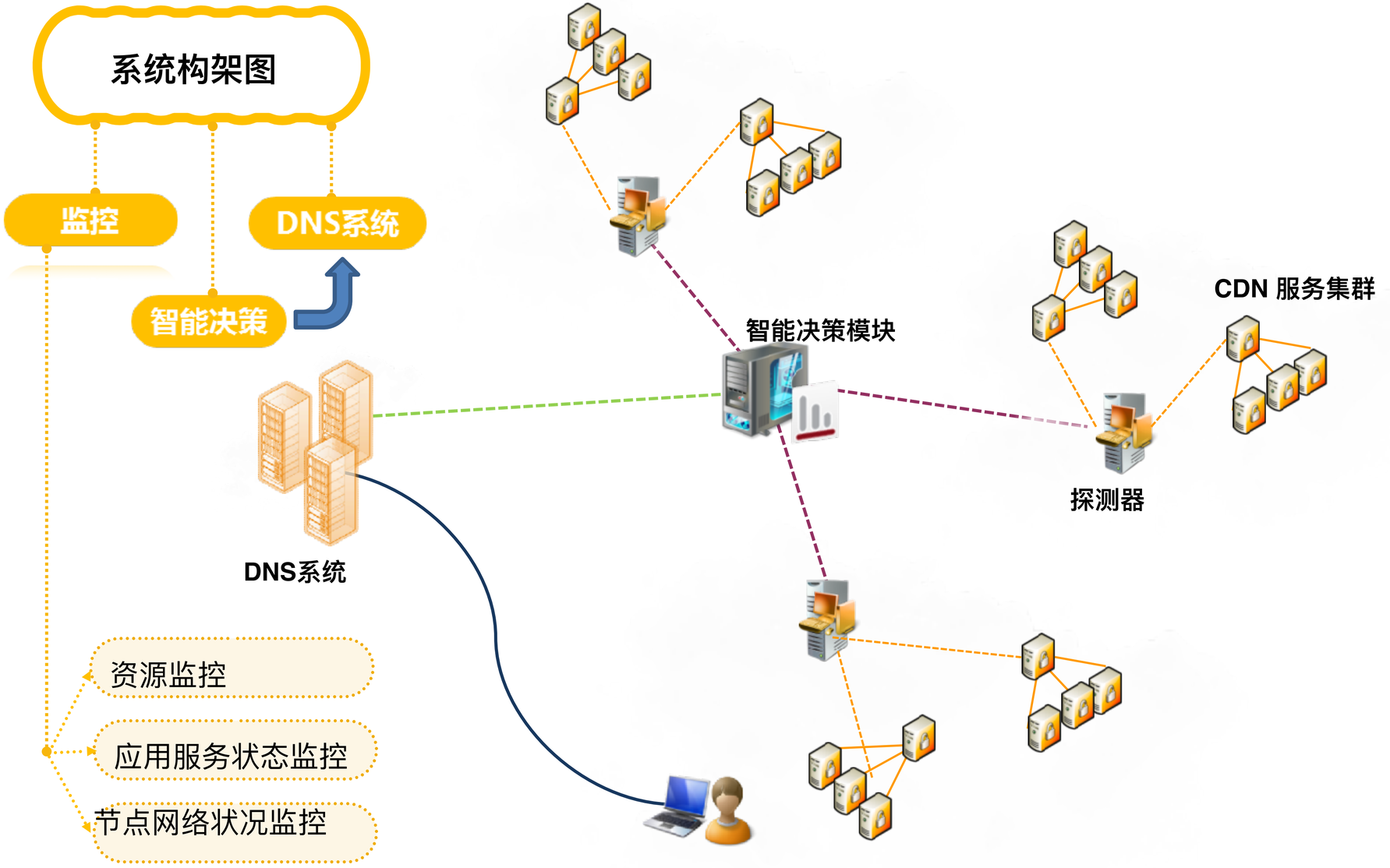
PART
02

通用优化篇

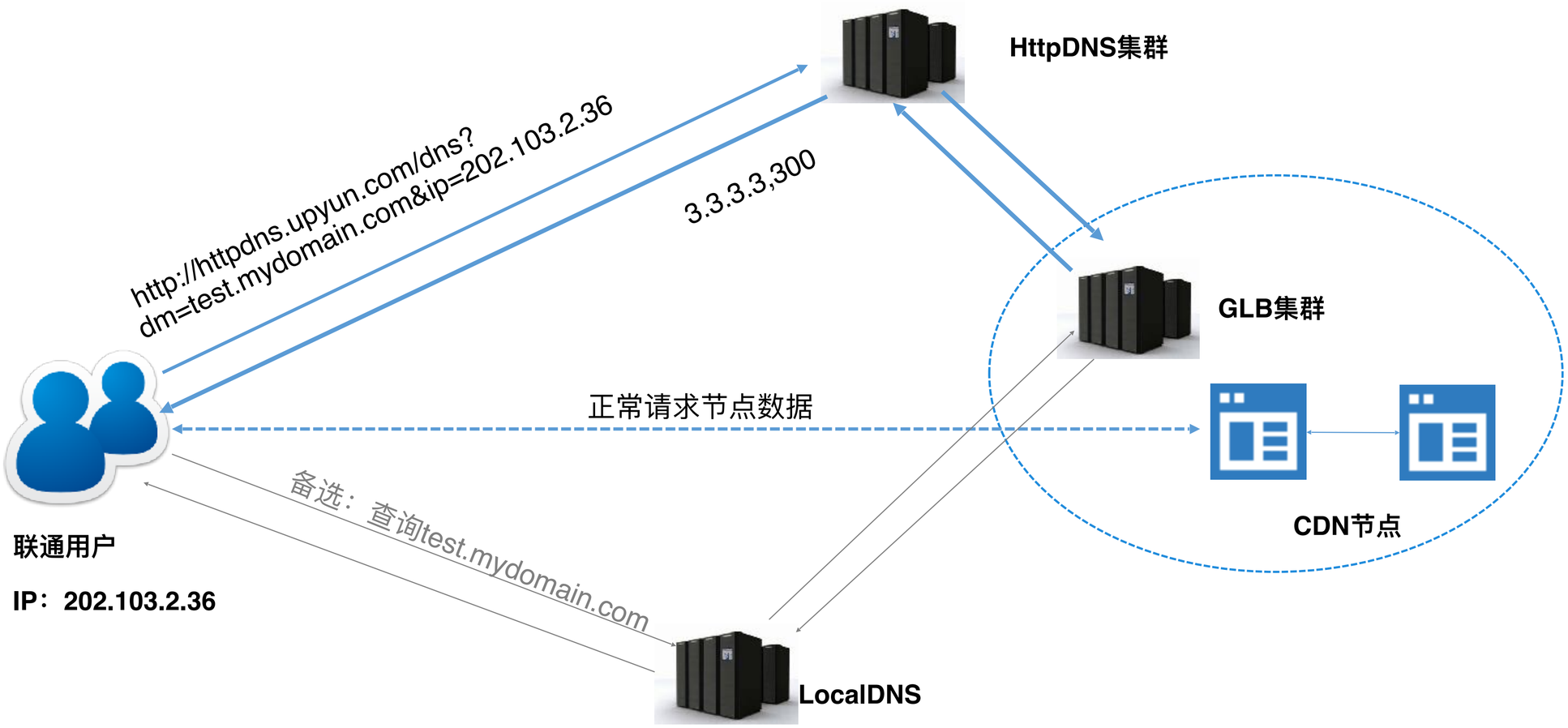
通用优化篇

- 智能调度：DNS 调度 + HTTP 302 调度 + HttpDNS 调度
- 内容优化：Gzip 压缩、JS/CSS 合并、代码压缩等
- 智能路由：动态中转、区域回源、回源负载均衡（A/B 测试）
- 缓存命中率：多域名共享缓存、过滤参数、缓存过期
- 协议优化：TCP 拥塞控制、HTTP/2

全局智能调度系统



HttpDNS 调度



HttpDNS访问原理图

PART
03

图片、视频篇

图片、视频优化篇

- 动态 URL 作图、样式作图
- WebP 自适应无痛接入方案
- H.265 点播自适应解决方案
- 根据视频头拖拉，支持 mp4、flv
- 视频大文件分片技术

动态 URL 作图 + 样式作图

实例二：https://p.upyun.com/docs/cloud/demo.jpg!upyun520

示例一：：https://p.upyun.com/docs/cloud/demo.jpg!/format/webp

WebP 自适应解决方案

基于 CDN 的 WebP 自适应图片无痛接入方案

图片流量更少，渲染速度更快

什么是 WebP

图片大小：138.1K

图片尺寸：583 x 328

图片大小：30.7K ⬇78%

图片尺寸：583 x 328

图片大小：385.4K

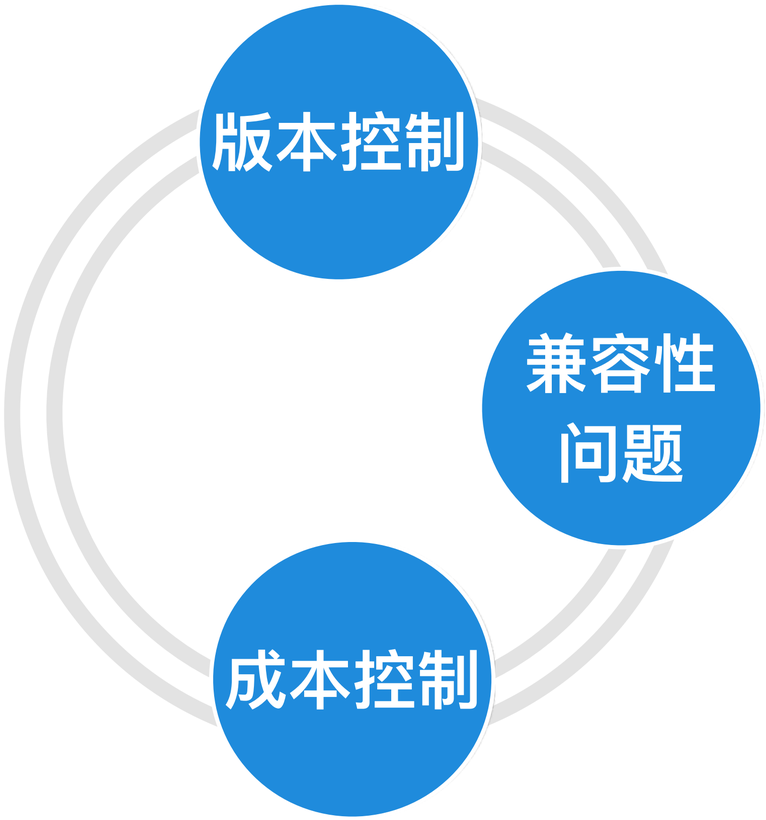
图片尺寸：583 x 328

图片大小：8.3K ⬇98%

图片尺寸：583 x 328

WebP 是一种支持无损压缩和有损压缩的网络图片格式，具有体积小、图片质量好，支持Alpha 透明以及24-bit 颜色数的特点，根据 Google 的测试，无损压缩后的 WebP 比 PNG 文件少了 45% 的文件大小，即使这些 PNG 文件经过其他压缩工具压缩之后，WebP 还是可以减少 28% 的文件大小。

WebP 自适应技术实现遇到的挑战

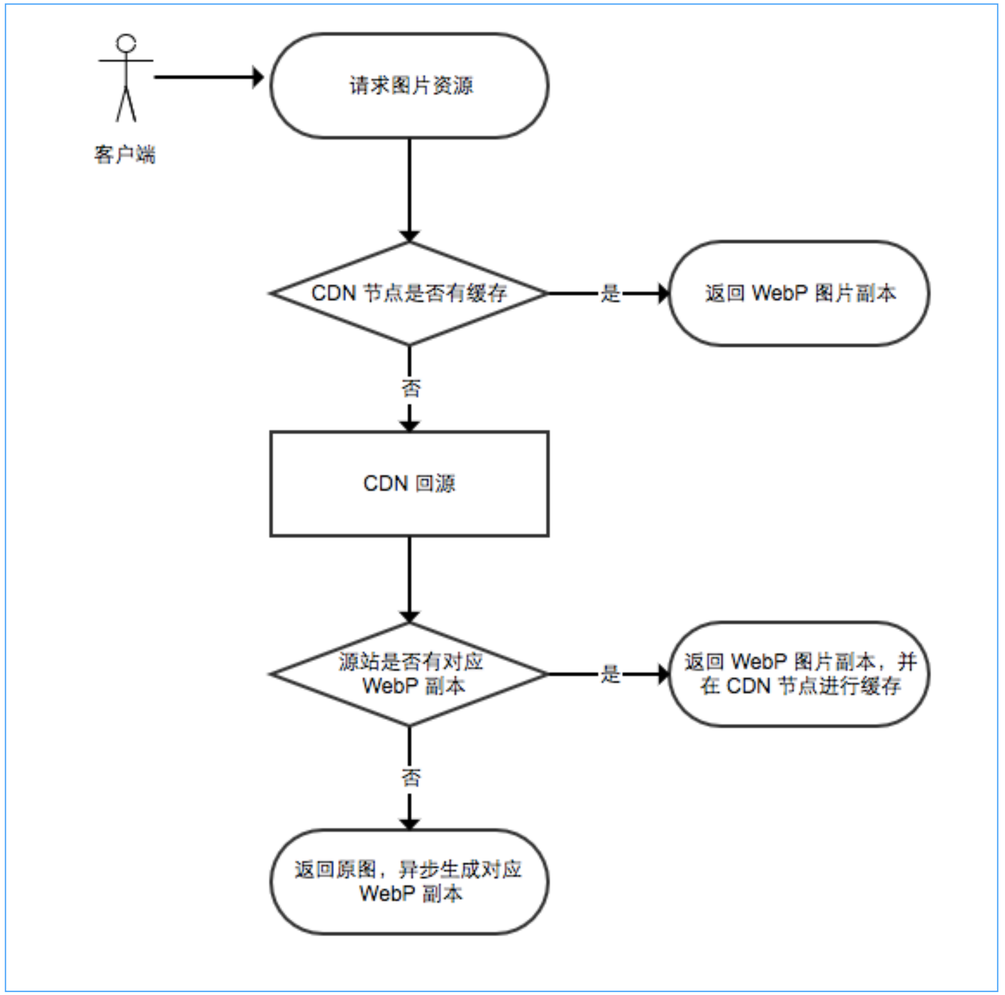


CDN 如何实现相同 URL 访问，缓存不同副本图片，减少源站改造成本。

并不是所有的浏览器都支持 WebP，需要解决浏览器兼容适配问题。

针对已经上线运营的网站，所有图片都替换成 WebP 格式，工作量巨大。

WebP 自适应技术实现方案



业务流程图

支持格式

类型	值
输入格式	JPG、JPEG、PNG、WebP、动态 WebP、GIF、动态 GIF、BMP、SVG等
输出格式	JPG、PNG、WebP、动态 WebP

支持有损、无损、动态 GIF 转换

关心的问题

- WebP 自适应的正确使用姿势?
- 无损 WebP 的正确使用姿势？

WebP 自适应技术优势

免费开放/弹性处理

WebP 自适应技术免费开放给平台所有用户使用。弹性处理集群，能够做到资源自动处理

一键开启

一键开启，即可完成 WebP 图片格式自适应转换，同时解决浏览器兼容性问题。

速度提升 50%

传输的文件大小减少，使得页面加载速度提升 50% 以上

成本降 30%

CDN 传输带宽减少，可有效节省 CDN 成本 30% 左右。

又拍云

WebP 自适应演示环节

https://huaban.com/all/

H.265 点播自适应解决方案

基于 CDN 的 H.265 自适应无痛接入方案

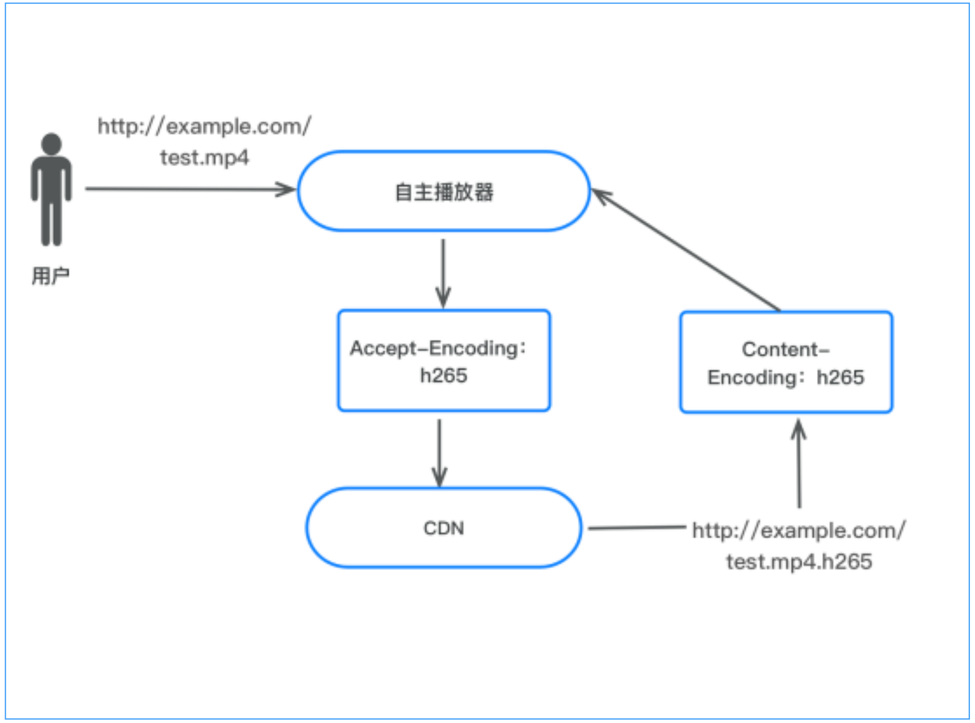
什么是 H.265

体积减少 50%+

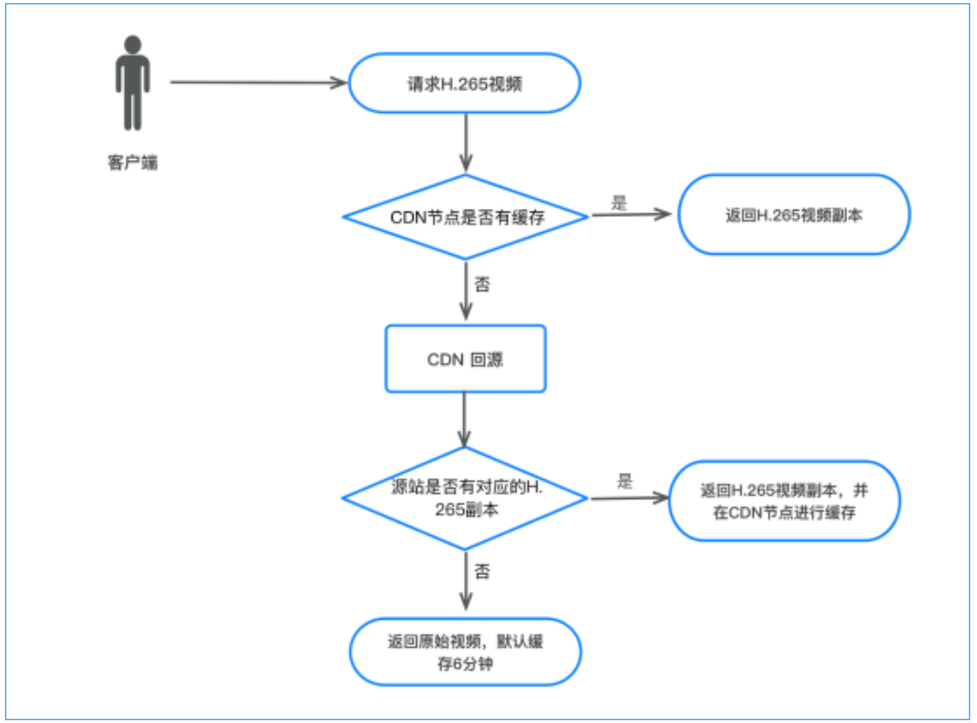
质量保持不变

H.265 又称 HEVC（High Efficieny Video Coding），是国际标准化组织和国际电联组织在 2013 年 3 月正式批准通过的新一代视频压缩标准，主要面向高清数字电视以及视频编解码系统的应用。H.265与 H.264 相比，拥有 2 倍的压缩效率 。

H.265 点播自适应实现方案

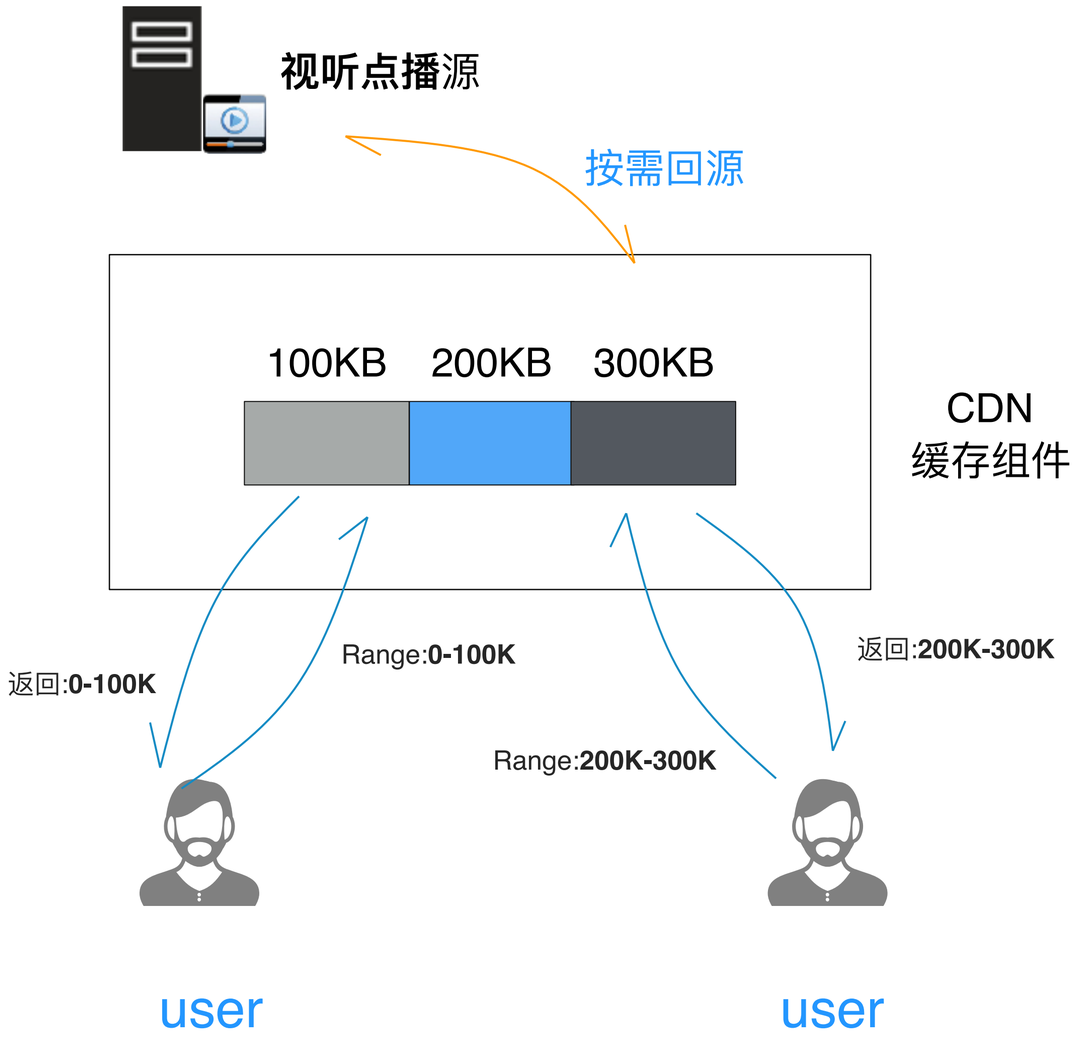


访问示意图

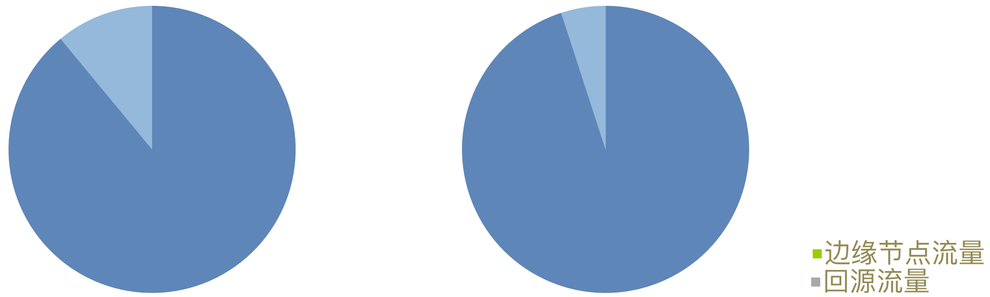


业务流程图

视频分片技术实现机制+分片预加载



某点播类客户，采用分片技术前后对比分析：



开启前　　开启后

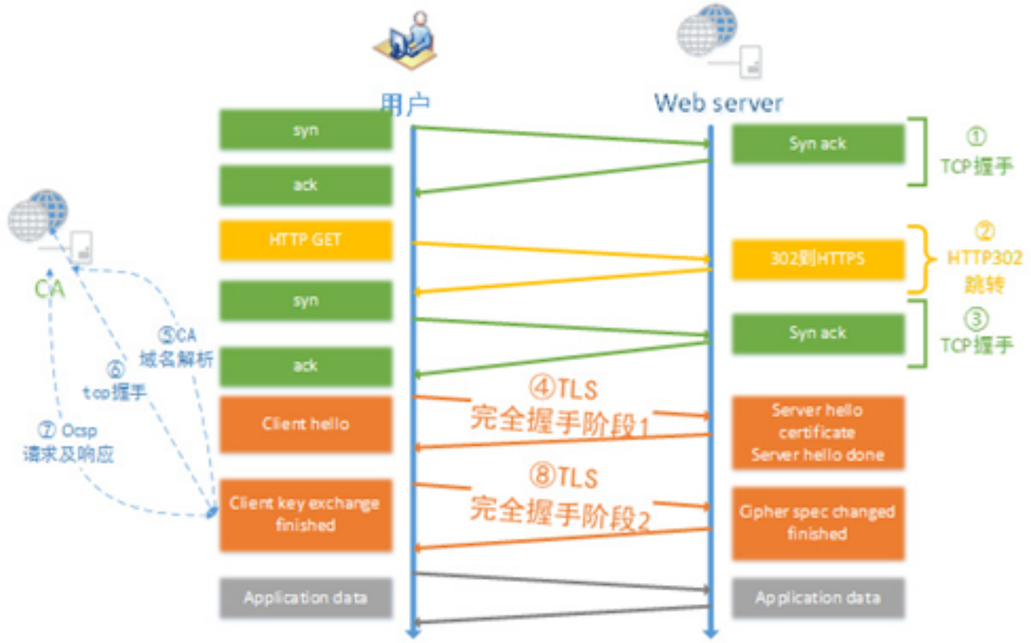
回源量从 **11%** 减少到 **5%**

缓存命中率从 **88%** 提升到 **95%**

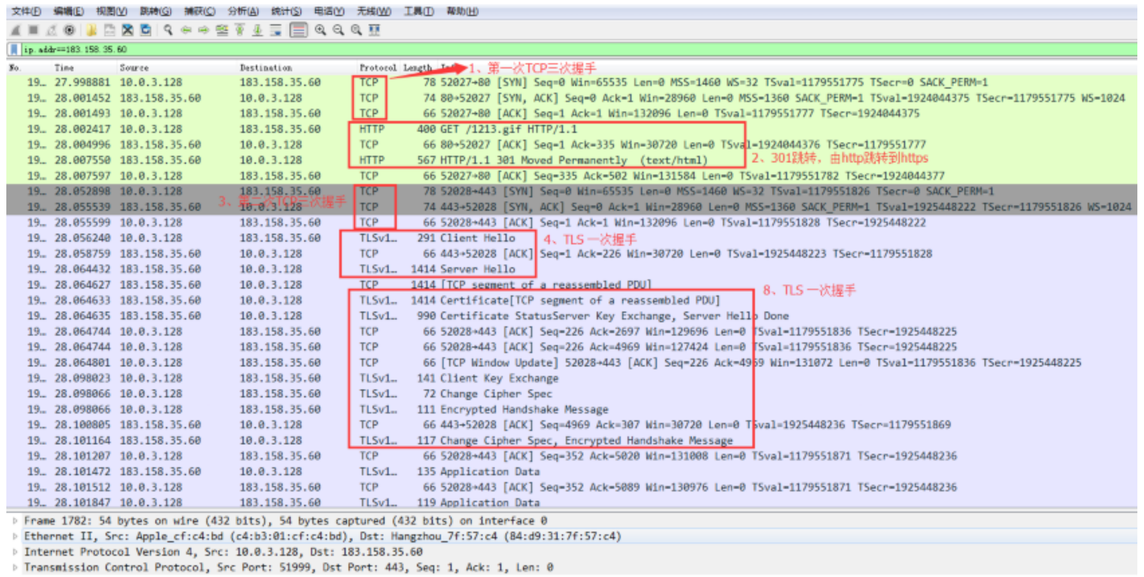
PART
04

HTTPS 篇

HTTPS 图解及消耗分析



HTTPS 业务流程



HTTPS 抓包

升级 HTTPS 遇到的挑战：慢 & 贵

- SSL 证书费用高及更新维护复杂
- HTTPS 降低用户访问速度（多次握手）
- HTTPS 消耗 CPU 资源，需要增加大量机器
- 由 HTTP 跳转到 HTTPS 的方式增加了用户访问耗时

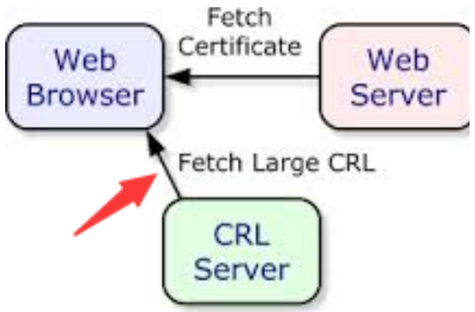
HTTPS 性能优化点

- HTTP/2
- OCSP Stapling
- 分布式 Session Cache(Session Resumption)
- HTTP Strict Transport Security(HSTS)
- False Start（依赖于PFS、NPN、ALPN ）
- 即将支持：Session ticket、Chacha20-Poly1305、TLSv1.3

协议升级：HTTP/2 支持

- 主流浏览器以及 ATS 强制使用 HTTPS
- 内容协商
  - NPN（Next Protocol Negotiation）
  - ALPN（Application Layer Protocol Negotiation

简化握手：OCSP Stapling




```
19… 28.064635 183.158.35.60   10.0.3.128     TLSv1…  990 Certificate StatusServer Key Exchange, Server Hello Done
19… 28.064744 10.0.3.128      183.158.35.60  TCP      66 52028→443 [ACK] Seq=226 Ack=2697 Win=129696 Len=0 TSval=117
19… 28.064744 10.0.3.128      183.158.35.60  TCP      66 52028→443 [ACK] Seq=226 Ack=4969 Win=127424 Len=0 TSval=117

▾ Secure Sockets Layer
  ▾ TLSv1.2 Record Layer: Handshake Protocol: Certificate Status
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 1623
    ▾ Handshake Protocol: Certificate Status
        Handshake Type: Certificate Status (22)
        Length: 1619
        Certificate Status Type: OCSP (1)
      ▾ Certificate Status
          Certificate Status Length: 1615
        ▾ OCSP Response
            responseStatus: successful (0)
          ▾ responseBytes
              ResponseType Id: 1.3.6.1.5.5.7.48.1.1 (id-pkix-ocsp-basic)
            ▾ BasicOCSPResponse
              ▾ tbsResponseData
                ▾ responderID: byKey (2)
                    byKey: a5043b7efb76a45a30637e2530837d361637f724
                  producedAt: 2017-03-26 07:17:23 (UTC)
                ▾ responses: 1 item
                  ▾ SingleResponse
                    ▷ certID
                    ▷ certStatus: good (0)
                      thisUpdate: 2017-03-26 07:17:23 (UTC)
                      nextUpdate: 2017-04-02 07:17:23 (UTC)
```

CRLs

ng

简化握手：分布式 Session Cache

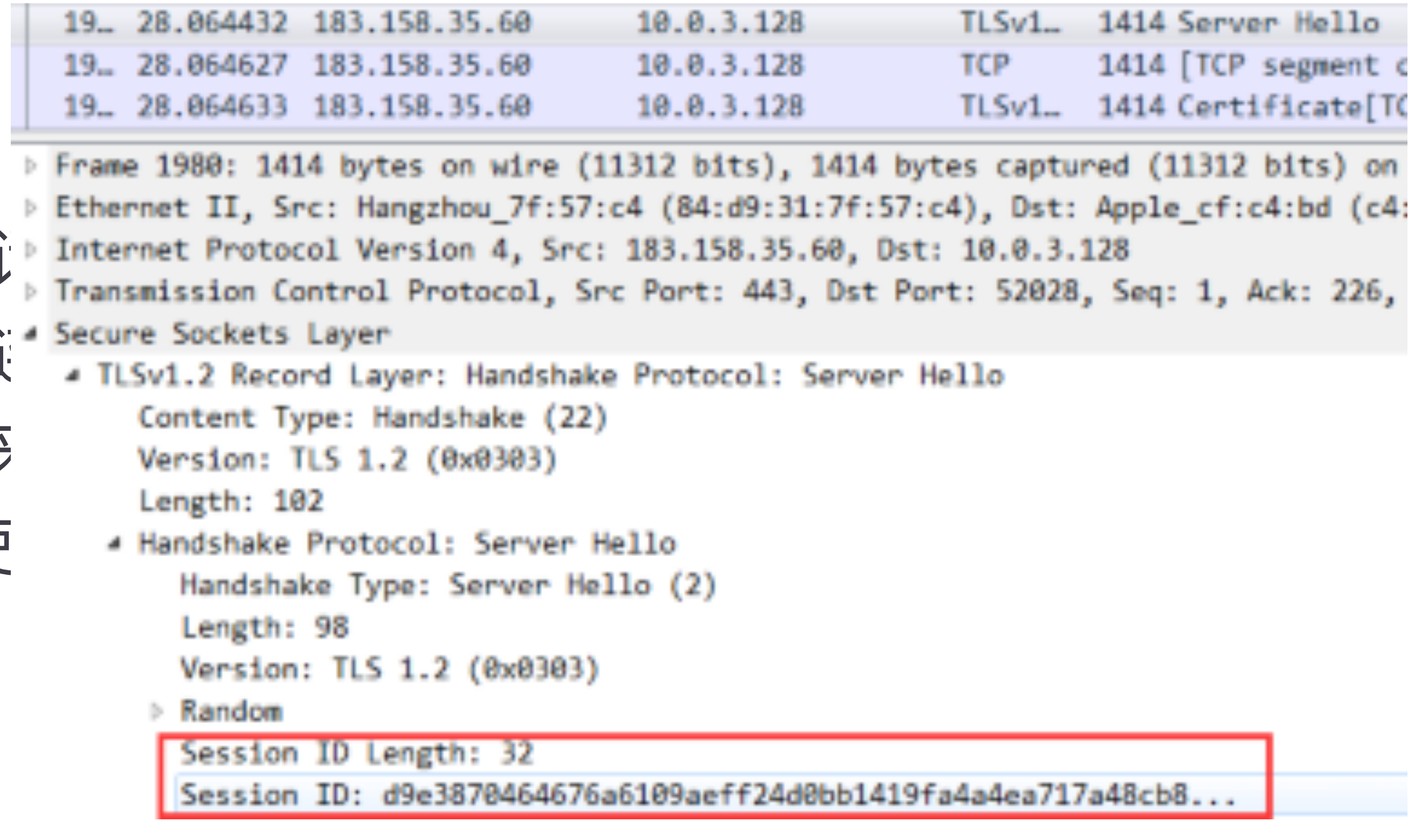
交互过程：

1）服务器为每一次的会话

2）如果客户端发起重新连

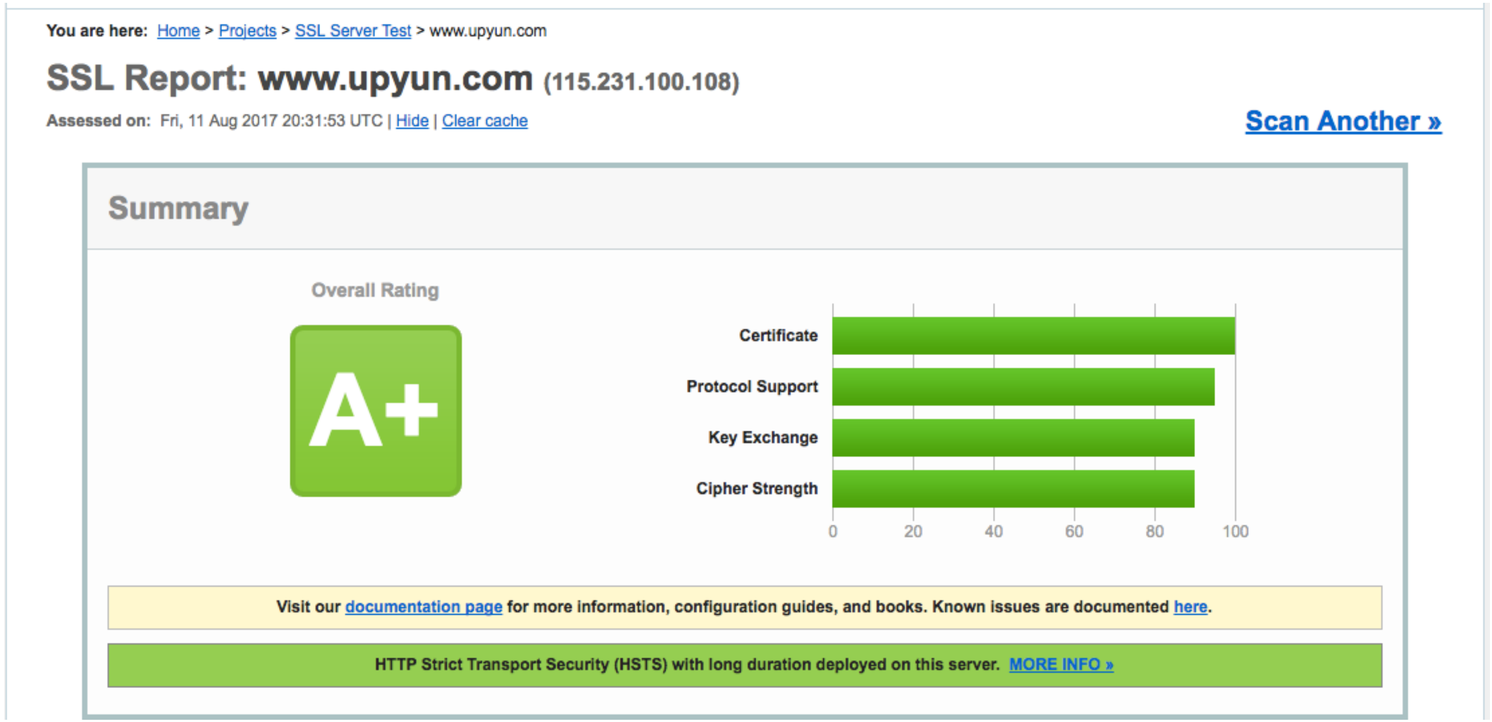
3）服务器收到客户端发来

之后，双方就可以重新使



网站 HTTPS 安全等级评分

- https://www.ssllabs.com/ssltest/



谢谢聆听！