

## HTTPDNS原理

**Tianny**

高级 SRE 工程师

[关注他](#)

8 人赞同了该文章

聊聊 HTTP DNS 的工作方式。

### 传统 DNS 存在的问题

当我们发出请求解析 DNS 的时候，首先，会先连接到运营商本地的 DNS 服务器，由这个服务器帮我们去整棵 DNS 树上进行解析，然后将解析的结果返回给客户端。但是本地的 DNS 服务器，作为一个本地导游，往往有自己的“小心思”。

一个最令人头痛的问题，相信每个人都遇到，就是域名劫持。

### HTTPDNS 概念

HTTPDNS 不走传统的 DNS 解析，而是自己搭建基于 HTTP 协议的 DNS 服务器集群，分布在多个地点和多个运营商。当客户端需要 DNS 解析的时候，直接通过 HTTP 协议进行请求这个服务器集群，得到就近的地址。

这就相当于每家基于 HTTP 协议，自己实现自己的域名解析，自己做一个自己的地址簿，而不使用统一的地址簿。但是默认的域名解析都是走 DNS 的，因而使用 HTTPDNS 需要绕过默认的 DNS 路径，就不能使用默认的客户端。使用 HTTPDNS 的，往往是手机应用，需要在手机端嵌入支持 HTTPDNS 的客户端 SDK。

么样的尴尬。

## HTTPDNS 工作模式

在客户端的 SDK 里动态请求服务端，获取 HTTPDNS 服务器的 IP 列表，缓存到本地。随着不断地解析域名，SDK 也会在本地图存 DNS 域名解析的结果。

当手机应用要访问一个地址的时候，首先看是否有本地的缓存，如果有就直接返回。这个缓存和本地 DNS 的缓存不一样的是，这个是手机应用自己做的，而非整个运营商统一做的。如何更新、何时更新，手机应用的客户端可以和服务器协调来做这件事情。

如果本地没有，就需要请求 HTTPDNS 的服务器，在本地图存 HTTPDNS 服务器的 IP 列表中，选择一个发出 HTTP 的请求，会返回一个要访问的网站的 IP 列表。

请求的方式是这样的。

```
curl http://106.2.xxx.xxx/d?dn=c.m.163.com
```

```
{"dns":[{"host":"c.m.163.com","ips":["223.252.199.12"],"ttl":300,"http2":0}],"client":
```

手机客户端自然知道手机在哪个运营商、哪个地址。由于是直接的 HTTP 通信，HTTPDNS 服务器能够准确知道这些信息，因而可以做精准的全局负载均衡。

## HTTPDNS 缓存设计

HTTPDNS 将解析速度和更新速度全部掌控在自己手中。

一方面，解析的过程，不需要本地 DNS 服务递归的调用一大圈，一个 HTTP 的请求直接搞定，要实时更新的时候，马上就能起作用；

另一方面为了提高解析速度，本地也有缓存，缓存是在客户端 SDK 维护的，过期时间、更新时间，都可以自己控制。

HTTPDNS 的缓存设计策略也是咱们做应用架构中常用的缓存设计模式，也即分为客户端、缓存、数据源三层，分别对应 SDK 客户端、本地缓存、HTTPDNS 服务器。

Redis是基于内存的缓存，但是同样提供持久化的能力，使得重启或者主备切换的时候，数据不会完全丢失。

SDK 中的缓存会严格按照缓存过期时间，如果缓存没有命中，或者已经过期，而且客户端不允许使用过期的记录，则会发起一次解析，保障记录是更新的。

解析可以同步进行，也就是直接调用 HTTPDNS 的接口，返回最新的记录，更新缓存；也可以异步进行，添加一个解析任务到后台，由后台任务调用HTTPDNS 的接口。

## HTTPDNS调度设计

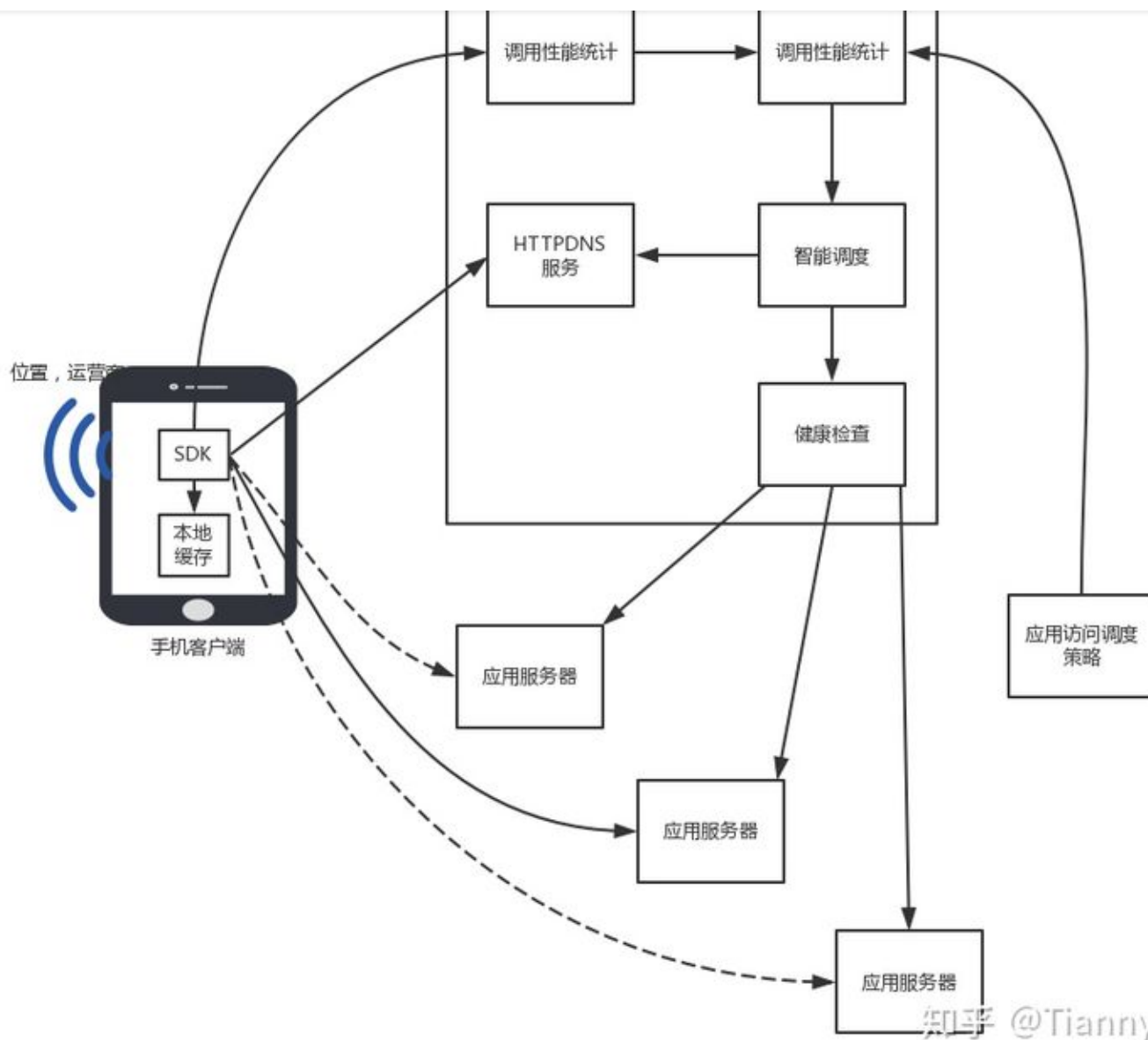
在客户端，可以知道手机是哪个国家、哪个运营商、哪个省，甚至哪个市，HTTPDNS服务端可以根据这些信息，选择最佳的服务节点返回。

如果有多个节点，还会考虑错误率、请求时间、服务器压力、网络状况等，进行综合选择，而非仅考虑地理位置。当有一个节点宕机或者性能下降的时候，可以尽快进行切换。

要做到这一点，需要客户端使用 HTTPDNS 返回的 IP 访问业务应用。客户端的 SDK 会收集网络请求数据，如错误率、请求时间等网络请求质量数据，并发送到统计后台，进行分析、聚合，以此查看不同的 IP 的服务质量。

在服务端，应用可以通过调用 HTTPDNS 的管理接口，配置不同服务质量的优先级、权重。HTTPDNS 会根据这些策略综合地理位置和线路状况算出一个排序，优先访问当前那些优质的、时延低的 IP 地址。

HTTPDNS 通过智能调度之后返回的结果，也会缓存在客户端。为了不让缓存使得调度失真，客户端可以根据不同的移动网络运营商 WIFI 的 SSID 来分维度缓存。不同的运营商或者 WIFI 解析出来的结果会不同。




## 小结

- 传统的 DNS 有很多问题，例如解析慢、更新不及时。因为缓存、转发、NAT问题导致客户端误会自己所在的位置和运营商，从而影响流量的调度。
- HTTPDNS 通过客户端 SDK 和服务端，通过 HTTP 直接调用解析 DNS 的方式，绕过了传统 DNS 的这些缺点，实现了智能的调度。

编辑于 2020-10-31

「真诚赞赏，手留余香」

文章被以下专栏收录



地表最强计算机网络串讲

计算机网络面试必看

推荐阅读



**freedns-go：一个无需配置，自动选择上游dns的工具**

姜晨耀

还没有评论

写下你的评论...

