

# Web JS SDK 性能说明

## Web JS SDK 加载

GrowingIO 发布的 Web JS SDK 在 CDN 上，CDN 服务商在各地部署有服务器，用户请求资源时就近获取所需内容，降低网络拥塞，提高用户访问响应速度和命中率。

因此，用户在加载网站请求 Web JS SDK 资源时，取决于 CDN 服务的响应速度和用户自身所在的网络环境。如果出现浏览网站时加载卡顿，响应时间过长，在排除 CDN 服务商的因素后排查自身网络。

## Web JS SDK 运行

Web JS SDK 运行在浏览器中，会占用部分内存进行监听及数据处理。需要使用到 Cookie 进行存放少量的用户 Id 等信息，存取率极低。SDK 上线时我们会经过多轮不同场景的测试，保证客户在使用过程中不会有运行问题出现。

## Web JS SDK 上报数据

Web JS SDK 上报方式主要有 3 种，分别为 sendBeacon、XMLHttpRequest、图片请求。

sendBeacon 介绍请见 [https://developer.mozilla.org/zh-CN/docs/Web/API/Beacon\\_API](https://developer.mozilla.org/zh-CN/docs/Web/API/Beacon_API)。不阻塞请求，不会对网站的业务请求有影响。除 IE 浏览器之外的其他主流浏览器均支持。

XMLHttpRequest 是为不支持 sendBeacon 的浏览器和 sendBeacon 上报失败的降级使用方式。是 IE 浏览器的主要上报方式。图片请求是对 XMLHttpRequest 请求失败时最后兜底的上报方案。尽最大可能地保证上报数据不会丢失。

上报数据服务端收到即可，所有的上报接口均没有返回值，状态码为 200 或 204。

上报时机：由于 Web 端产生的数据不会进行存储，因此我们会实时发送（即事件产生即上报）。

上报频率：取决于页面事件触发频率。

## CDN 平均时延测试

我们通过模拟用户使用场景专门对 SDK 的 CDN 加载进行了测试。

## 测试环境：



## 测试项目描述：

- 1、新建一个没有任何节点（即 Body 为空）的 Html 文件，在 Head 中仅引入 Web SDK CDN 地址（即）  
以上是为了排除其他资源和 Dom 加载的影响，在加载网页时仅有 html 和 SDK 的 js 两个请求，以便准确获取 SDK 加载时间。
- 2、新建本地 Node 服务。
- 3、新建 Jest 测试用例。使用 puppeteer 进行浏览器可视化测试。大致测试逻辑如下（单次）：
  - 1) puppeteer 创建浏览器实例（打开 Chromium）
  - 2) 获取页面实例，添加请求监听（监听一个请求发起和结束）。在监听中判断 script 类型的请求，发起请求时获取当前时间戳，结束请求时获取当前时间戳并与开始时间相减获得该请求从开始到返回的时长（即 SDK 加载总时长）。由于是监听方式，会存在与浏览器之间的细微误差，但误差值小于 10ms。
  - 3) 获得 SDK 加载总时长后打印日志存入数组并关闭浏览器。
  - 4) 重复上述测试过程 50 次，并对数据进行求平均值。
- 4、由于命令行无法显示过多日志结果，因此需进行多次手动启动，获取到用例中的平均值后再取平均值。重复操作 10 次，因此请求测试次数即为 500 次。

### 测试结果：

经过 GrowingIO 单轮 50 次，重复 10 轮，即共 500 次的测试结果，在请求 CDN 上的 SDK 加载的时延平均约为 202ms。

测试结果仅供参考，结果受网络环境条件影响大。

附部分截图：

```
console.log
  打开空白页面 48 加载 SDK, 获取请求

    at test/index.spec.js:22:13
      at runMicrotasks (<anonymous>)

console.log
  请求时长: 260 ms

    at test/index.spec.js:35:17
      at Array.map (<anonymous>)

console.log
  打开空白页面 49 加载 SDK, 获取请求

    at test/index.spec.js:22:13
      at runMicrotasks (<anonymous>)

console.log
  请求时长: 207 ms

    at test/index.spec.js:35:17
      at Array.map (<anonymous>)

console.log
  打开空白页面 50 加载 SDK, 获取请求

    at test/index.spec.js:22:13
      at runMicrotasks (<anonymous>)

console.log
  请求时长: 182 ms

    at test/index.spec.js:35:17
      at Array.map (<anonymous>)

console.log
  请求次数: 50

    at test/index.spec.js:48:11

console.log
  平均请求时长: 199.06 ms

    at test/index.spec.js:49:11
```

```
PASS test/index.spec.js (75.507 s)
测试 SDK 加载速度
```



```
console.log
  打开空白页面 48 加载 SDK, 获取请求

    at test/index.spec.js:22:13
    at runMicrotasks (<anonymous>)

console.log
  请求时长: 181 ms

    at test/index.spec.js:35:17
    at Array.map (<anonymous>)

console.log
  打开空白页面 49 加载 SDK, 获取请求

    at test/index.spec.js:22:13
    at runMicrotasks (<anonymous>)

console.log
  请求时长: 177 ms

    at test/index.spec.js:35:17
    at Array.map (<anonymous>)

console.log
  打开空白页面 50 加载 SDK, 获取请求

    at test/index.spec.js:22:13
    at runMicrotasks (<anonymous>)

console.log
  请求时长: 207 ms

    at test/index.spec.js:35:17
    at Array.map (<anonymous>)

console.log
  请求次数: 50

    at test/index.spec.js:48:11

console.log
  平均请求时长: 214.68 ms

    at test/index.spec.js:49:11
```

```
PASS test/index.spec.js (75.211 s)
测试 SDK 加载速度
```

# SDK 加载慢问题排查

## 如何排查是否为 CDN 服务的问题

使用全国网络测试的工具类网站进行检测。

参考地址：

<https://tool.chinaz.com/speedtest/https://assets.giocdn.com/2.1/gio.js>

CDN 资源时延受地域影响较大。一般可能会有部分节点存在无响应或延迟高的问题，但不影响资源的正常加载。因为 CDN 服务有负载均衡及时延算法，在加载资源时除了优先就近，还会考虑时延，从而提供最优地址进行资源下发。

附：全国 2022 年 3 月 17 日 18:00 瞬时时延。

监测点	解析IP	HTTP状态	总耗时	解析时间	连接时间	下载时间	文件大小	下载速度
内蒙古呼和浩特(电信)	61.147.211.209	200	135ms	37ms	42ms	56ms	36.38KB	985.19KB
广东深圳(电信)	14.21.76.71	200	267ms	9ms	7ms	251ms	36.37KB	498.13KB
湖北十堰(电信)	58.220.72.83	200	710ms	25ms	20ms	665ms	36.41KB	187.32KB
内蒙古呼和浩特(电信)	49.71.74.18	200	130ms	35ms	39ms	56ms	36.38KB	1023.08KB
安徽合肥(电信)	150.138.101.76	200	55ms	10ms	17ms	28ms	36.37KB	2418.18KB
湖北十堰(电信)	49.79.232.33	200	83ms	21ms	22ms	40ms	36.36KB	1602.41KB
广东佛山(电信)	14.21.76.72	200	9074ms	16ms	9037ms	21ms	36.38KB	14.66KB
四川成都(电信)	118.123.233.23	200	20ms	6ms	2ms	12ms	36.40KB	6650KB
内蒙古呼和浩特(电信)	58.220.72.83	200	125ms	36ms	38ms	51ms	36.39KB	1064KB
江西南昌(电信)	182.106.174.17	200	9061ms	1ms	9022ms	38ms	36.40KB	14.68KB
江西吉安(电信)	111.206.176.78	200	28ms	<1ms	7ms	21ms	36.37KB	4750KB
内蒙古呼和浩特(电信)	58.220.72.83	200	117ms	35ms	35ms	47ms	36.37KB	1136.75KB
辽宁锦州(电信)	58.220.72.83	200	149ms	51ms	39ms	59ms	36.41KB	892.62KB
福建福州(电信)	117.24.11.18	200	44ms	20ms	6ms	18ms	36.40KB	3022.73KB
云南昆明(电信)	61.147.211.209	200	152ms	46ms	46ms	60ms	36.37KB	875KB
内蒙古呼和浩特(电信)	58.220.72.83	200	124ms	37ms	36ms	51ms	36.39KB	1072.58KB
江苏徐州(电信)	49.79.232.33	200	58ms	23ms	12ms	23ms	36.37KB	2293.10KB
浙江宁波(电信)	115.223.11.149	200	70ms	35ms	12ms	23ms	36.40KB	1900KB
安徽滁州(电信)	49.71.73.132	200	46ms	7ms	10ms	29ms	36.37KB	2891.30KB
江苏徐州(多线)	49.71.73.132	200	62ms	24ms	12ms	26ms	36.40KB	2145.16KB
江苏徐州(多线)	61.147.211.209	200	176ms	134ms	16ms	26ms	36.37KB	755.68KB
浙江绍兴(多线)	223.113.149.62	200	262ms	236ms	9ms	17ms	36.39KB	507.63KB
江苏徐州(多线)	49.79.232.33	200	69ms	26ms	11ms	32ms	36.37KB	1927.54KB
河南新乡(多线)	61.163.111.20	200	238ms	203ms	11ms	24ms	36.38KB	558.82KB
江苏徐州(多线)	49.71.74.18	200	62ms	17ms	10ms	35ms	36.38KB	2145.16KB
河南新乡(多线)	58.220.72.83	200	86ms	35ms	17ms	34ms	36.39KB	1546.51KB
河南新乡(多线)	111.206.176.78	200	395ms	48ms	48ms	299ms	36.37KB	336.71KB
江苏徐州(多线)	49.71.73.132	200	107ms	77ms	10ms	20ms	36.37KB	1242.99KB
河南新乡(多线)	61.147.211.209	200	114ms	37ms	22ms	55ms	36.39KB	1166.67KB
江苏徐州(多线)	49.71.74.18	200	58ms	24ms	10ms	24ms	36.39KB	2293.10KB
河南洛阳(多线)	49.71.73.132	200	70ms	16ms	22ms	32ms	36.39KB	1900KB
四川成都(联通)	222.132.143.108	200	121ms	39ms	34ms	48ms	36.39KB	1099.17KB
天津(联通)	117.12.41.16	200	140ms	130ms	1ms	9ms	36.37KB	950KB
山东济南(联通)	60.211.170.13	200	35ms	11ms	4ms	20ms	36.39KB	3800KB
黑龙江哈尔滨(联通)	139.215.252.13	200	45ms	8ms	17ms	20ms	36.37KB	2955.56KB
四川成都(联通)	60.211.170.13	200	134ms	38ms	34ms	62ms	36.39KB	992.54KB
安徽合肥(联通)	112.28.237.61	200	142ms	64ms	22ms	56ms	36.39KB	936.62KB
江苏镇江(联通)	222.132.143.108	200	68ms	31ms	13ms	24ms	36.41KB	1955.88KB
四川成都(移动)	223.113.149.62	200	127ms	40ms	37ms	50ms	36.39KB	1047.24KB
安徽合肥(移动)	112.28.235.150	200	16ms	2ms	2ms	12ms	36.39KB	8312.5KB

## 如何排查是否为自身网络问

使用代理软件、网络带宽低、计算机内存过小、计算机中毒等等。  
排查方式一般为更换网络环境、更换设备重试对比排查等。

## 如何降低 CDN 时延对资源加载的影响

可尝试将 SDK 文件下载至本地项目中，部署在业务服务器上，可以有效降低 CDN 服务可能的不稳定导致的时延过长问题。