

如果一个 h5 页面很慢该如何排查

先确定和加载慢还是渲染慢。

通过 chrome devtools 排查

如果是加载慢，可以通过 network 面板查看所有资源加载时间，找到加载慢的那个资源进行优化。

如果是渲染慢，可以通过 performance 面板查看，

- First Paint (FP) 开始渲染。是页面导航与浏览器将该网页的第一个像素渲染到屏幕上所用的中间时
- First Contentful Paint (FCP) 第一次有内容
- DOM Content Loaded (DCL) DOM 渲染完成
- Largest Contentful paint (LCP) 最大的资源渲染完成
- Loaded 所有资源渲染完成。

分析每一步所花时间，排查具体渲染慢的那一步。

通过 lighthouse 第三方工具访问

- 先全局安装 lighthouse

```
npm install lighthouse -g
```

在输入需要分析的网站的地址。

```
// --view 可视化查看，--preset=desktop 在桌面查看  
lighthouse https://www.baidu.com --view --preset=desktop
```

解决方法

加载慢

- 优化服务器硬件配置，使用 cdn。
- 路由懒加载，大组件异步加载，减少包体积。
- 优化 http 缓存策略。

渲染慢

- 优化服务端接口（ajax 请求获取数据慢）
- 优化前端组件内部的逻辑。
- 使用 SSR 服务端渲染。

解决慢是逐步优化升级解决，不像解决 bug，一次性解决。持续跟进统计结果，分析性能瓶颈，持续优化。