

高频事件(scroll,mousemove等)的触发频繁触发回调导致的大量计算会引发页面的抖动甚至卡顿

节流防抖结合

节流

防抖

时间中的节流(throttle)与防抖(debounce)

React: <https://juejin.im/post/5b506ae0e51d45191a0d4ec9>

应用层性能优化

首屏优化体验

图片lazyload

首屏只展示首屏展示的图片，当下拉加载时再去请求真实图片，减少性能的压力

实现思路

1常规方案监听onscroll判断图片是否在可视区

关键点

- 1.当前可视区域的高度 `const viewHeight = window.innerHeight || document.documentElement.clientHeight`
- 2.该元素距离可视区域顶部的高度：`dom.getBoundingClientRect().top`

实现：

2IntersectionObserver API(可实现懒加载和无限滚动)

API使用：http://www.ruanyifeng.com/blog/2016/11/intersectionobserver_api.html

polyfill:<https://polyfill.io/v3/polyfill.min.js?features=IntersectionObserver>

实现：

预渲染骨架屏

直出一个骨架HTML，挂载DOM之后再替换

PWA: 淘宝是利用Workbox 缓存的骨架屏