



链滴

# 前端 VR 开发技术栈选型

作者: [HerbertHe](#)

原文链接: <https://ld246.com/article/1645912023128>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 写在前面的

一直想做VR很久了，之前没做的原因如下：一、没钱买单反及相关器材；二、完全没有游戏开发背景三、技术水平有限.....给 Vditor PR 一直在等待合并，最近的工作不是很忙，然后开始继续做前端VR发。

自己是个业余的风光摄影爱好者，尼康好兄弟！目前手上的器材是：尼康 Z5 + 24-70mm f/4 Z S-lin + 123全景云台 PH7 + 铭匠 11mm f/2.8 鱼镜头（在路上）+ 捷宝853pro三脚架，参考花费接近.5w。

目前做VR主要是渲染自己合成拍摄的VR全景照片，然后再探索跨学科的3D模型渲染（本专业学的是用物理学。

因为之前完全没有游戏开发经验，一来我就硬刚了 Three.js。看到渲染器渲染场景，场景里面有物体物体又由几何构型 + 材质组成等等内容的时候，就很头大。Three.js 代码一行行添加物体，这之后项的维护难度很大，处理的细节也很多。

不可否认，Three.js 的功能非常的强大。现阶段在前端VR开发上，主流的方案是 AFrame 或者 Three.s，及类似项目或衍生项目。

## 技术栈选型探索

因为自己主要是做 React 的开发，前端开发框架自然会偏向选择 React。但在 VR 开发框架上，我更向 AFrame，因为 Three.js 的开发太过抽象；对新手来说，上手难度太大、细节也存在处理不到位的问题。先调研了 React + AFrame 的方案，发现了一个帖子指出了 React Diff 会导致 AFrame 渲染性低的问题。

我自己结合 React Diff 的基本原理，思考了一个非常重要的问题：如果将 AFrame 封装为通用组件通过修改 props 进行渲染会导致什么问题？假设外部传入的 props 发生了改变，React 必然会触发 AFrame 封装组件重新渲染，而 AFrame 场景的实体整体被 React 重新渲染又会触发 AFrame 对于 VR 场景的重新渲染。这严重违背了节省性能开销的初心！AFrame 整体框架的重新渲染会使性能降低，React Diff 做了不必要的组件渲染。如果使用 Three.js 封装组件供 React，正常情况下并不会出现这个问题，可以复用实例化的 Three.js 实例。

所以，我觉得 React + Three.js 是一个最合适的技术栈选型搭配。

但为了减少自己在 VR 开发方向优化的心智负担，我并没有选择 React + Three.js 的方案。因为 AFrame 出色的性能优化和完善的新手教程、文档，让我可以为此更换前端框架进行开发。回到上面的那问题，如果不做组件的整体渲染，那么我的需求就得到满足了。根据自己熟悉的框架的原理，我可以择 Vue、也可以选择 Svelte 进行开发。但是最后我选择了 Svelte，因为 Vue 对于 TypeScript 还在上；并且作为多语言开发者，我更喜欢更少的API。

因此，我最终选择的技术栈是 Svelte + AFrame。当然，还有用到一些其它的开源项目；为了敏捷开，我另外选择了 Vite + WindiCSS 的组合，可以迅速启动我的开发。

## 技术栈选型总结

我认为下面两种组合是很棒的：

- React + Three.js
- Svelte + AFrame

上面的技术栈并不太适合新手，但非常适合进阶开发。如果是前端小白的话，非常不建议采用这套方案。React + Three.js 的技术学习路线，可能会直接让你放弃前端；如果是有几年的前端开发经验上手 VR 开发，建议第二套方案，可以敏捷开发并且实现完全够用的性能，Svelte 的上手也不是很难。

不过之后我还是会向第一套方案进行探索，实现高定制化、高性能的前端 VR 开发。