

全部开发者教程

17：深入事件更新：vue event invokers

18：框架实现：ELEMENT 节点下，事件的挂载和更新

19：局部总结：ELEMENT 节点的挂载、更新、props 打补丁等行为总结

20：源码阅读：renderer 渲染器下，Text 节点的挂载、更新行为

21：框架实现：renderer 渲染器下，Text 节点的挂载、更新行为

22：源码阅读：renderer 渲染器下，Comment 节点的挂载行为

23：框架实现：renderer 渲染器下，Comment 节点的挂载行为

24：源码阅读：renderer渲染器下，Fragment 节点的挂



Sunday • 更新于 2022-10-19

← 上一节 21：框架实现：... 23：框架实现：... 下一节 →

22：源码阅读：renderer 渲染器下，Comment 节点的挂载行为

完成了 Text 节点的挂载、更新之后，那么下面我们来看 Comment 节点的挂载操作。

这里大家要注意：vue 不支持动态的 comment，即没有更新操作

其实对于 Comment 而言，它的整体流程和 Text 非常类似。

我们来看一下，创建测试实例 packages/vue/examples/imooc/runtime/render-comment.html：

<> 代码块

```
1 <script>
2   const { h, render, Comment } = Vue
3
4   const vnode = h(Comment, 'hello world')
5   // 挂载
6   render(vnode, document.querySelector('#app'))
7 </script>
```

跟踪代码实现：

1. 进入 patch 方法，case Comment，触发 processCommentNode 函数
1. 进入 processCommentNode 函数：
2. 内部的代码非常简单，只触发了两个方法：
1. hostCreateComment：内部的代码在 packages/runtime-dom/src/nodeOps.ts 中的 createComment 方法：
- <> 代码块
- ```
1 createComment: text => doc.createComment(text),
```
- 创建 Comment 节点
2. hostInsert：这个在 Text 节点时，刚看过，这里就不在查看了

至此，Comment 节点渲染成功。

由以上代码可知：

1. 对于 Comment 而言，只存在 挂载 操作
2. 整体的处理非常简单：
1. 通过 doc.createComment 创建 Comment 节点
2. 通过 hostInsert 完成挂载



