Sunday • 更新于 2022-10-19

上一节 17：深入事件更... 19：局部总结：... 下一节

18：框架实现：ELEMENT 节点下，事件的挂载和更新

本小节我们就来实现事件的更新和挂载操作。

1. 在 `packages/runtime-dom/src/patchProp.ts` 中，增加 `patchEvent` 事件处理

代码块

```
    } else if (isOn(key)) {
        // 事件
        patchEvent(el, key, prevValue, nextValue)
    }
```

2. 在 `packages/runtime-dom/src/modules/events.ts` 中，增加 `patchEvent`、`parseName`、`createInvoker` 方法：

代码块

```
/**
 * 为 event 事件进行打补丁
 */
export function patchEvent(
    el: Element & { _vei?: object },
    rawName: string,
    prevValue,
    nextValue
) {
    // vei = vue event invokers
    const invokers = el._vei || (el._vei = {})
    // 是否存在缓存事件
    const existingInvoker = invokers[rawName]
    // 如果当前事件存在缓存，并且存在新的事件行为，则判定为更新操作。直接更新 invoker 的 va
    if (nextValue && existingInvoker) {
        // patch
        existingInvoker.value = nextValue
    } else {
        // 获取用于 addEventListener || removeEventListener 的事件名
        const name = parseName(rawName)
        if (nextValue) {
            // add
            const invoker = (invokers[rawName] = createInvoker(nextValue))
            el.addEventListener(name, invoker)
        } else if (existingInvoker) {
            // remove
            el.removeEventListener(name, existingInvoker)
            // 删除缓存
            invokers[rawName] = undefined
        }
    }
}

/**
 * 直接返回剔除 on，其余转化为小写的事件名即可
 */
function parseName(name: string) {
    return name.slice(2).toLowerCase()
```

```
/**
 * 生成 invoker 函数
 */
function createInvoker(initialValue) {
    const invoker = (e: Event) => {
        invoker.value && invoker.value()
    }
    // value 为真实的事件行为
    invoker.value = initialValue
    return invoker
}
```

支持事件的打补丁处理完成。

可以创建如下测试实例 `packages/vue/examples/runtime/render-element-event.html`：

代码块

```
<script>
  const { h, render } = Vue

  const vnode = h('button', {
    onClick() {
      alert('点击')
    },
  }, '点击')
  // 挂载
  render(vnode, document.querySelector('#app'))

  setTimeout(() => {
    const vnode2 = h('button', {
      // onClick() {
      //   alert('click')
      // },
      onDblclick() {
        alert('双击')
      },
    }, '双击')
    // 挂载
    render(vnode2, document.querySelector('#app'))
  }, 2000);
</script>
```

测试成功

17：深入事件更新：vue event invokers ◂ 上一节　　下一节 ▸ 19：局部总结：ELEMENT 节点的挂载、更...

我要提出意见反馈

企业服务　网站地图　网站首页　关于我们　联系我们　讲师招募　帮助中心　意见反馈　代码托管

意见反馈　收藏教程　标记书签