



链滴

关于 solo 中使用 vditor 的几个疑问

作者: [lisaem](#)

原文链接: <https://ld246.com/article/1564386986801>

来源网站: [链滴](#)

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

关联的问题, 在帖子 [Vditor 使用指南](#)中的一个评论

下面是原文

一个小问题 希望有缘人回答一下。本人在自己的服务器上部署了博客 solo, 在家用 vditor 编辑预览章都没问题, 今天来公司, 想写点东西, 发现预览功能无法使用, 左边输入的内容, 右边无法显示预览, 一片空白。chrome 里面用 f12 抓了下网络请求, 发现有个请求被公司内网屏蔽了, 具体是 我的地址 /console/markdown/2html, 这个请求失败导致的。我试了下给出的教程链接 <https://hacpai.com/guide/markdown> 这个预览就没问题, 我看网络也是请求到后台去的 <https://hacpai.com/markdown>, 然后返回解析后的 html 网页。

1. 想问下 目前 vditor 的预览不是用 markdown-it 支持前端预览吗? 我是个大后端, 对前端不明白, 为何这个预览还要网络请求, 貌似是调到后台的转换方法 flexmark 了, 还是有什么开关或者置?

2. vditor 本身是只是一个 markdown 编辑器, 转换预览 (解析器功能) 还是依赖第三方的组件, 端预览用 markdown-it, 集成在 vditor 中, 通过后端预览是发送到后端 (路径是 我的网址 /console/markdown/2html) 然后先判断

MARKDOWN_ENGINE_URL 是否可用, 否的话就用集成的 flexmark。我这样理解对不对?

针对以上问题又仔细看了下代码以及使用指南的相关评论, 目前预览中实现是这样

```
vditor.mdTimeoutId = window.setTimeout(() => {
  const renderStartTime = new Date().getTime();
  if (vditor.options.preview.url) {
    const xhr = new XMLHttpRequest();
    xhr.open("POST", vditor.options.preview.url);
    xhr.setRequestHeader("Content-Type", "application/json;charset=UTF-8");
    xhr.onreadystatechange = () => {
      if (xhr.readyState === XMLHttpRequest.DONE) {
        if (xhr.status === 200) {
          const responseJSON = JSON.parse(xhr.responseText);
          if (responseJSON.code !== 0) {
            alert(responseJSON.msg);
            return;
          }
          this.element.innerHTML = responseJSON.data;
          this.afterRender(vditor, renderStartTime);
        }
      }
    };

    xhr.send(JSON.stringify({
      markdownText: vditor.editor.element.value,
    }));
  } else {
    md2html(vditor, vditor.options.preview.hljs.enable).then((html) => {
      this.element.innerHTML = html;
      this.afterRender(vditor, renderStartTime);
    });
  }
}, vditor.options.preview.delay);
```

会去判断 vditor.options.preview.url, 而 solo 中使用 vditor 的时候是在 page.js 这个文件中设置了这个

局的变量的

```
window.vditor = new Vditor('soloEditorComment', {  
  
  placeholder: that.tips.commentContentCannotEmptyLabel,  
  height: 180,  
  tab: '\t',  
  hint: {  
    emojiPath: Label.staticServePath + '/images/emoji',  
  },  
  esc: function () {  
    $('#soloEditorCancel').click()  
  },  
  ctrlEnter: function () {  
    $('#soloEditorAdd').click()  
  },  
  preview: {  
    delay: 500,  
    show: false,  
    url: Label.servePath + '/console/markdown/2html',  
    hljs: {  
      enable: true,  
      style: Label.hljsStyle,  
    },  
    parse: function (element) {  
      if (element.style.display === 'none') {  
        return  
      }  
      Util.parseLanguage()  
    },  
  },  
  counter: 500,  
  resize: {  
    enable: resizeEnable,  
    position: 'top',  
  },  
  lang: Label.langLabel,  
  toolbar: toolbar,  
})  
vditor.focus()  
}
```

也就是现在每次预览都是去请求我后台的那个servePath + '/console/markdown/2html'的方法，而会用到vditor本身内置的markdown-it这个库了，不知道这样理解对不对。至于为何不能直接在前端预览，我也看到@88250的回复了，表示理解。另外，现在是不是B3log相关的产品都不在前端做预览，而是统一后端做，比如用你们推出的marked-http，vditor中加入这样的判断，是为了开源给大，让使用vditor的人自己选择用前端还是用后端做渲染，这样理解对吗？如果我要在solo中用前端渲染，是不是修改page.js中的url就可以了？感谢回答。