

# 错误监控

## 错误分类

- 代码错误。 `try...catch`, `window.onerror`
- 资源加载错误。比如图片 `image.onerror`, `performance.getEntries()`, 捕获阶段监听 `error` 事件。

`window.onerror` 只能捕获即时运行错误，对于资源运行错误没法捕获。`window.onerror` 只能捕获那些未被其他错误处理程序捕获的错误。也就是说，如果在代码中使用 `try...catch` 结构来捕获错误，那么 `window.onerror` 将不会被触发。

使用 `window.addEventListener('error', function(event) {})` 添加的错误监听器可以捕获到所有类型的错误，包括那些已经被其他错误处理程序捕获的错误。

```
// 获取加载的资源名称
// performance.getEntries() 返回一个数组
performance.getEntries().forEach(item => console.log(item.name))
```

```
// 捕获阶段监听 error 事件
// 错误监控代码写在加载错误资源的前面
<script>
    window.addEventListener("error", function(event) {
        console.log("捕获event", event);
    }, true)
</script>
<script src="https://www.baidu.com/xcvbnmkjhgf.html"></script> // 模拟一个没有的地址，模拟资源加载错误
```

## 跨域的资源如何监控

需要服务端配合了

- 客户端加上 `crossorigin`。
- 服务端加上 `Access-Control-Allow-Origin`。

## 上报错误的基本原理

- 通过 Image 对象上报， `(new Image()).src='xxxxxx'`，最简单也不需要借助第三方库。
- 通过 AJAX 通信上报。