对于JavaScript客户端产生错误的原因和后果的研究

IEEE学生成员：Frolin S. Ocariza Jr.，IEEE学生成员：Kartik Bajaj，IEEE高级成员：Karthik Pattabiraman和IEEE成员：Ali Mesbah

摘要——JavaScript客户端广泛用于Web应用程序，以改善用户交互并将客户端与服务端的通讯最小化。不过，众所周知，JavaScript容易出错。虽然先前的研究已经证明了JavaScript问题是普遍存在的，但没有尝试去确定它们的原因和后果。我们的研究目标是了解JavaScript错误的根本原因和影响，以及结果会如何影响JavaScript的程序员，测试人员和工具开发人员。我们对来自19个bug存储库的502个bug报告进行了实证研究。我们详细查看了bug报告，分类提取了每个bug原因（错误）和后果（失败和影响）的相关信息。我们的结果表明，大多数（68％）JavaScript错误与DOM相关，这意味着它们是由JavaScript代码与文档对象模型（DOM）的错误交互引起的。此外，80％最有影响的JavaScript错误与DOM相关。最后，大多数JavaScript错误都源于使用JavaScript编写代码的程序员的错误，而不是其他Web应用程序组件。这些结果表明JavaScript的程序员和测试人员需要能够帮助他们找出DOM的工具。此外，开发人员可以使用我们发现的错误模式为JavaScript设计更强大的静态分析工具。

索引术语——错误，JavaScript，文档对象模型（DOM），错误报告，实证研究

1 简介

WEB开发人员通常使用JavaScript来增强客户端上Web应用程序的交互性。例如：JavaScript可以将事件处理程序分配给不同的Web应用程序组件，例如：按钮、链接和输入框，从而在用户与组件进行交互时有效的定义Web应用程序的功能。此外，JavaScript可用于向服务器发送异步HTTP请求并更新响应的网页内容。

JavaScript包含了几个与传统语言有区别的功能。首先，JavaScript代码在异步模型下执行。这允许事件处理程序在用户与Web应用程序组件交互时按需执行。其次，许多JavaScript旨在与称为文档对象模型（DOM）的外部实体进行交互。此实体是一个动态树状结构，包括Web应用程序中的组件以及它们的组织方式。使用DOM API调用，JavaScript可用于访问或操作存储在DOM中的组件，从而无需重新加载页面来更改网页。

虽然上述功能允许Web应用程序具有较高的交互性，但它们还为JavaScript代码中的错误提供了其他途径。在之前的研究[1]中，我们从50个常用的Web应用程序中收集了JavaScript控制台的消息（即异常），来了解Web应用程序对形成JavaScript错误的容易程度以及这些应用程序中出现的JavaScript的错误类型。虽然这项研究指出了JavaScript的问题的普遍存在性，但我们没有探究它们的影响因素，也没有分析它们产生的问题类型。了解问题产生的根本原因和影响对于开发人员，测试人员以及工具开发人员来说至关重要，因为这可以提高应用程序的可靠性。

在本文中，我们的目标是发现Web应用程序中JavaScript的问题（错误）的原因，并分析其后果（失败和影响）。为实现这一目标，我们对500多份公开的JavaScript bug报告进行了实证研究。 我们选择的bug报告，它们通常有关于JavaScript bug的详细信息，并且还记录了Web应用程序的使用方式，这是难以从JavaScript的控制台消息或静态分析中提取出来的信息。 此外，我们将我们的搜集到的错误报告标记了“固定”，这样可以排除虚假或严重的错误报告。

然而，研究bug报告的一个重要挑战是，很少有Web应用程序公开它们的bug库。即便是有会这样做的人，也常常以特别的方式公开报告，这使得从报告[2]中提取出详细信息具有挑战性。因此，我们系统地收集错误报告，并规范它们的格式，以便更好的研究它们。

我们的主要工作有：

* 我们收集并系统地对15个Web应用程序和4个JavaScript库的502个bug报告进行分类，并将报告以标准格式进行归类；
* 我们将JavaScript的问题分为多个类别。我们认为有一个问题类别支配其他的问题，即DOM相关的JavaScript的问题（下面会有更多细节）；
* 我们分析有多少种错误可以被归类为类型错误，这有助于我们评估向JavaScript添加严格类型系统的编程语言的实用性，比如TypeScript[3]和Dart[4]；
* 我们定量分析它们的性质（即原因和后果）和JavaScript的问题产生的影响；
* 在适当的情况下，我们对我们分析的每个错误报告的特性进行了时间分析。分析的结果将表明这些年来的技术特性设置的变化趋势，来判断JavaScript代码的开发人员、测试人员和工具开发人员是否朝着提高JavaScript客户端的可靠性方向前进；

我们的研究结果表明，大约68%的JavaScript的问题与DOM相关，这是由于JavaScript代码和DOM之间的错误交互而产生的。举个简单的例子：使用不正确的ID来检索DOM元素，会导致null异常。此外，我们发现DOM的相关错误占Web应用程序中严重错误的80%左右。最后，我们认为，大多数错误是由于JavaScript代码而不是服务器端代码/HTML引起的，并且有些错误的重复编程导致的。