# 如何编写更佳的bug report

作者：[A8572785](http://my.csdn.net/A8572785" \t "http://lib.csdn.net/article/softwaretest/_blank)

**如何编写更佳的bug report**

**---** **MihirKamdar《HowTo** **Write Better Bug Reports》**

我们是否经常看到开发人员针对我们归档的bug report要求提供更多的信息？我们是否经常需要在bug report归档后花更多的时间去研究那个问题？我们是否经常从开发人员那里听到在他们那边难以重现bug并且需要即刻提供“可重现的步骤”？广义上来说，我们与其花更多的时间在这些问题上还不如投资更多的时间来测试系统。问题出在bug report的质量上。这里介绍一些如何改进并达到完美bug report的建议。

**Bug report的目的**

当我们发现一个缺陷时，我们需要把它告诉给开发人员。Bug report就是这种沟通的媒介物。Bug report的主要目的是让开发人员亲眼看到这个错误。如果你不能和他一起以在他面前制造出那个失败，那么就需要给他们足够多的指引以便他们能够自己制造出那个失败。Bug report就是解释在期望结果和实际结果之间的差距并且详细的说明如何重现那个场景。

**在发现缺陷之后**

·        只有当你确信你已经发现一个bug的时候开始起草bug report，不要在测试结束或每天结束之后。那样，你可能会遗忘掉一些东西。更糟的情况是，我们可能会忘掉那个bug。

·        花一些时间去诊断你正在报告的缺陷。想想可能存在的原因。可能到最后你会发现更多的缺陷。在你的bug report中说说你的发现。开发人员将不仅仅对你使他们的工作变得轻松而感到高兴。

·        在开始读你的bug report之前抽出一些时间来。你可能会感觉到象重新编写报告一样。

**摘要**

Bug report的摘要是你bug report给读者的第一印象。你提交的bug的命运很大程度依赖于你的bug report能否吸引读者。原则就是每个bug应该有一个简单有趣的摘要。它可能会听上去象编写一个优秀的勾起注意的广告活动。但是随后，没有什么意外。一个好的摘要应该不超过50到60个字符。而且一个好的摘要不应该承载任何对bug主观的表达。

**语言**

·        不要在bug report中夸大缺陷。同样，也不要太轻描淡写了。

·        不管bug是多么的令人讨厌，别忘了是bug令人讨厌，而不是开发人员。永远不要冒犯开发人员的努力。使用委婉些的说法。“混乱的UI”可以被温和些改为“不正确的UI”。这样开发人员的努力将会得到尊重。

·        保持简单诚实。你不是在写散文或文章，因此使用简单的语言

·        在编写bug report的时候记住你的目标读者。他们可能是开发人员，其他的测试人员，经理，或者在一些情况下，甚至是客户。Bug report应该可以被所有的人理解。

**可重现的步骤**

·        “可重现的步骤”的流程应该是合乎逻辑的。

·        清楚的列出前提条件

·        写下平常的步骤。例如，如果一个步骤要求用户创建文件并且为它命名，不要要求用户命名为“Mihir’sfile”。最好命名为好像“Test File” 一样的文件名。

·        “可重现的步骤”应该详尽。例如，如果你想用户在MicrosoftWord里保存一个文件，你可以要求用户到File菜单并且点击Save子菜单项。你也可以只说“保存文件”。但是记住，并不是所有的人都知道如何在MicrosoftWord中保存文件。因此最后遵守第一种方法。

·        在一个干净的系统里测试你的“可重现的步骤”。你可能会发现有些步骤被遗漏或是毫无关系的。

**测试数据**

尽力编写普通的bug report，开发人员可能没有权限访问你的测试数据。如果bug是和一组特定的测试数据相关，在你的bug report上附带上它。

**截屏**

截屏是bug report中一个十分必要的部分。一个图片胜过一千句话。但是不要把在每个bug report里附带没有必要的截屏变成一个习惯。理想的来说，你的bug report应该是足够有效的使开发人员重现问题。截屏应该只是验证的一种方法。

·        如果你要在bug report里附带截屏，要确保那些图片不是太大的，使用jpg或gif的格式，而不是bmp格式

·        在截屏上写上注释以指出问题所在。这将帮助开发人员一眼就可以马上定位问题。

**严重程序/优先级别**

·        在设置bug report的严重程序之前应该全面的分析缺陷的影响程序。如果你认为你的bug具有很高的优先级应该被修复，在bug report中证明这点。应该在bug report的描述部分指出这个理由。

·        如果bug是来自上个内部小版本或版本回归的结果，那么发出警报。象这种bug的严重程序可能是低的，但是优先级别应该是高。

**日志**

在bug report里附上日志或日志的摘录片断。这将帮助开发人员轻松地分析且调试系统。多数情况下，如果不附上日志而且在开发人员那边又很难重现问题的话，他们将会把bug report打回给你并要求附带日志文件。

如果日志文件不太大的话，举个例子，大约20到25行，你就可以把它贴在bug report里。但是如果它比较大的话，把它做为附件贴在bug report里，否则你的bug report会看上去象个日志。

**其他信息**

·        如果你的bug是随机出现的，只需在你的bug report中说一下就可以了。但是不要忘记归档它。你总是能够在你发现它们之后的任何时间里增加准确的步骤。这也将在其他人提交这个问题时解救你，特别是当那个问题比较严重时。

·        在bug report中写下错误信息，特别是当错误信息有编号的时候。例如，来自数据库中的错误信息。

·        在bug report中写下版本编号和内部小版本编号

·        写下问题可以被重现的平台。准确的说明问题不可重现的平台。同样也要理解问题在特定平台上不可重现和没有在某个平台上测试之间的分别。这个可能会造成混淆。

·        如果你遇到几个问题却有一样的结果，只需写一个bug report。问题的修复可能只是一个。同样，如果你在不同的地方遇到相似的问题，且要求同一种修复方法,但是在不同的地方，那么就要为每一个问题书写单独的bug report。每个bug report对应一个修复。

·        如果可以重现bug的测试环境是开发人员可以访问的，写下访问这种设置的详情。这将帮助他们节约安装环境的时间以重现你提交的bug。

·        你决不能坚持关于bug的任何信息。在bug被修复之前由于低效的提交bug而引起的开发人员和测试人员之间不必要的交互只是浪费时间