遇到一个诡异 Bug，每逢周三就崩溃 - 文章 - 伯乐在线

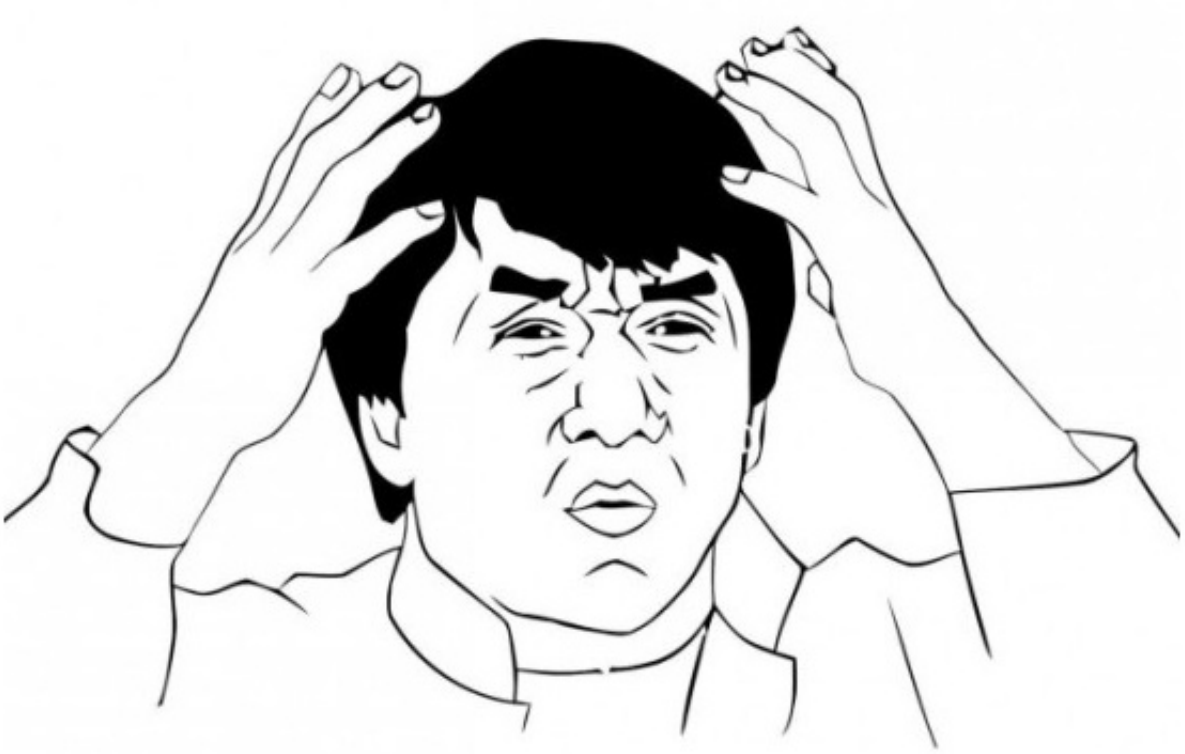
拿点儿喝的坐好，是时候讲讲我最喜欢的 bug 的故事了。

那是我第一份 IT 相关的工作：在一个生产重要医疗设备的厂商担任软件开发的暑期实习生。那些设备主要是麻醉给药系统和病患监控设备，后者就是在卧床患者旁边放着的发出“哔哔”声的那种盒子，上面会以图形方式显示患者的脉搏、血压、呼吸等等。如果心电图变成一条直线的话还会立刻召唤护士。当时的办公室里全是 2 米高的装着笑气的罐子，还有长着超级大胡子的嵌入式系统大拿，整屋子的人都在给各种设备准备文档，为了让它们通过 FDA 的认证。时不时还有人小声提到 10 年前没能在测试中发现的一个 bug，它导致了一台麻醉机在手术过程中间重启了。不用说，对于像我这种十几岁的新手，所有的生产系统肯定是不会让我们碰的。

（伯乐在线补注：一氧化二氮（Nitrous Oxide），又称笑气，无色有甜味气体，是一种氧化剂，化学式 N_2O，在一定条件下能支持燃烧（同氧气，因为笑气在高温下能分解成氮气和氧气），但在室温下稳定，有轻微麻醉作用，并能致人发笑。）
不过他们还是给我安排了一份让人羡慕的工作，去测试一个在 1997 年听起来还十分时髦的原型项目：一个用 C++ 编写的服务器，它会监听患者监控设备的串口，然后把一些需要关注的事件转存到 SQL Server 数据库中，之后通过 CORBA 把数据发送到 Java Applet，于是医生或者相关人员就能通过互联网看到这个患者的状态了，它既能看到实时的数据，也能浏览之间的数据记录。帅气！只是那个时候我对这些语言和系统都一无所知！

接下来的几个星期就像杀猪一样的折腾，主要时间都花在了读懂让人头疼的 Visibroker ORB 手册，还有超级普通的类型转换 bug，不过我终于让我的“辛普森”系统磕磕绊绊地跑起来了，它用“Homer”（注：辛普森一家里的老爸）来记录和提供数据，然后用“Bart”（注：辛普森一家里的熊孩子）来进行显示。这几个星期让我觉得 CORBA 复杂得让人想死、AWT 让人头疼欲裂（比如 GridBagLayouts，呕）、applet 慢得像只蜗牛，不过 Java 看起来倒还像是个挺不错的语言。不过还有个小麻烦：C++ 服务器时不时就会突然崩溃掉，然后我开始尝试去搞明白到底是为什么。

因为我监听的那台监控设备在另一间屋子里，所以我绝大部分的开发和测试都是通过手动的“演示”模式来完成的，比如在一个循环里模拟一次心脏停跳之类的，据我所知，我的服务器从来没在这个过程中宕机过。不过在我或者别人手动摆弄那些控制器的时候，它确实崩溃过，尤其是在实际机器上操作的时候，不过我想尽办法也没能找到一个方法能让它稳定重现，甭管怎么做都不行。我把所有事件日志都记录到磁盘上，想找到在崩溃之前到底发生了什么，不过我小心翼翼地按照事件序列精确地手动重复了每一次事件（比如：把过滤器设置为 X，把控制器旋钮向右拧三个刻度，点击按钮……），我在两间屋子里跑来跑去（因为我在摆弄患者监控设备的时候是看不见我电脑上的日志的），但始终都没能让崩溃重现。不管是什么“鬼事件”（对我就是这么叫它的），它肯定是在造成崩溃的同时还逃过了所有日志。是不是有什么串口 I/O 或者硬件问题中断了事件？难道是宇宙射线把我 PC 上的数据位给改变了？



我把整天整天的时间都用来尝试去重现这个错误，但是毫无结果，在经历了几个星期的挫折之后，我最后干脆在所有从串口收到事件和写入数据库的操作中间都加了 printf 语句，在这个过程中，我重新检查了每一行代码，然后终于逐渐见到了曙光。

当我创建数据库结构的时候，为了节省空间而犯了一个错误，一个新手常犯的错误：把时间戳当成主键了。所以如果两个事件在一个毫秒内发生的话，数据库就会抛出主键唯一性约束的异常（译注：SQL Server 的 datetime 类型的精度其实不是1毫秒，而是3.33毫秒）。我之前注意到这个问题了，不过我觉得这种情况非常罕见，而且只会在没那么重要的环境中发生（比如在鼓捣监控设备内部配置的时候），所以我只是加了个 catch 语句，在日志中写了一条警告信息，然后继续执行后面的操作。

但是！这是个老派的代码，记录日志使用 C 语言风格的代码编写的，把日志字符串记录到了一个长度为 80 个字符的缓冲区中。唯一性异常这个消息本身是个常量，而日志的时间戳是格式化的，也就是实用了完整的英文的星期拼写（%E），所以输出就类似于“Monday, July 17, 1997, 10:38:47.123”。最后就是因为英文里面星期几的拼写有个有意思的属性：

星期几	单词长度
Sunday	6
Monday	6
Friday	6
Tuesday	7
Thursday	8
Saturday	8

Wednesday	9

明白了吧？星期三（Wednesday），而且只在星期三的时候，如果有人在监控器配置那儿手动进行了一个特定操作的话，就会在同一毫秒内产生两个事件，于是导致数据库抛出异常，而这个异常的消息包括字符串结尾的终结符的话，则刚刚好 81 个字符，导致了 80 个字符的缓冲区溢出，把程序搞挂了！

在那之后，在所有需要使用的数据库表中，我都会确保去用一个专门的、自增的整数 ID 作为主键，然后用 ISO 格式（也就是 YYYY-MM-DD）而不是星期几来记录所有日志。这些年来，我学到了不管一个 bug 看上去多么随机和不可预测，如果你挖得足够深的话，总是能找到一个符合逻辑的解释，极少有真的“不相关”的错误，几乎都是你特么自己的错。