Mars Code

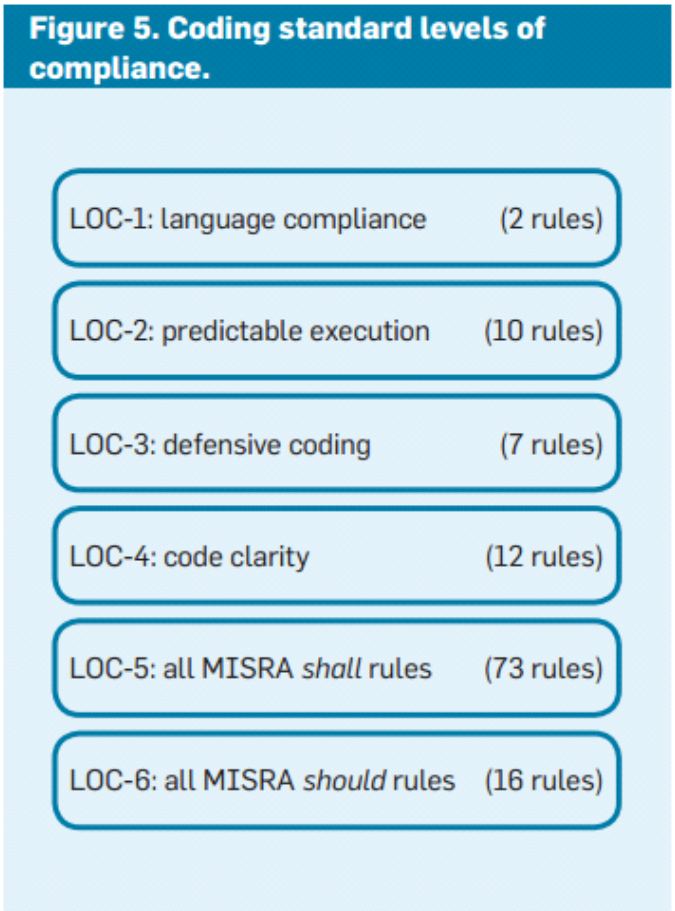
Monday, March 21, 2016 9:28 PM

这篇文章主要描述了前往火星的探测器好奇者号上的Code是如何保证其正确性的

好奇者号上的嵌入式系统，是为了一系列不通用的外围设备所设计的，用户只有一个任务，有些关键代码可能只会使用一次，而且很难测试。极小的错误可能导致非常严重的后果。因此，如何减少错误和风险对好奇者号的开发至关重要。

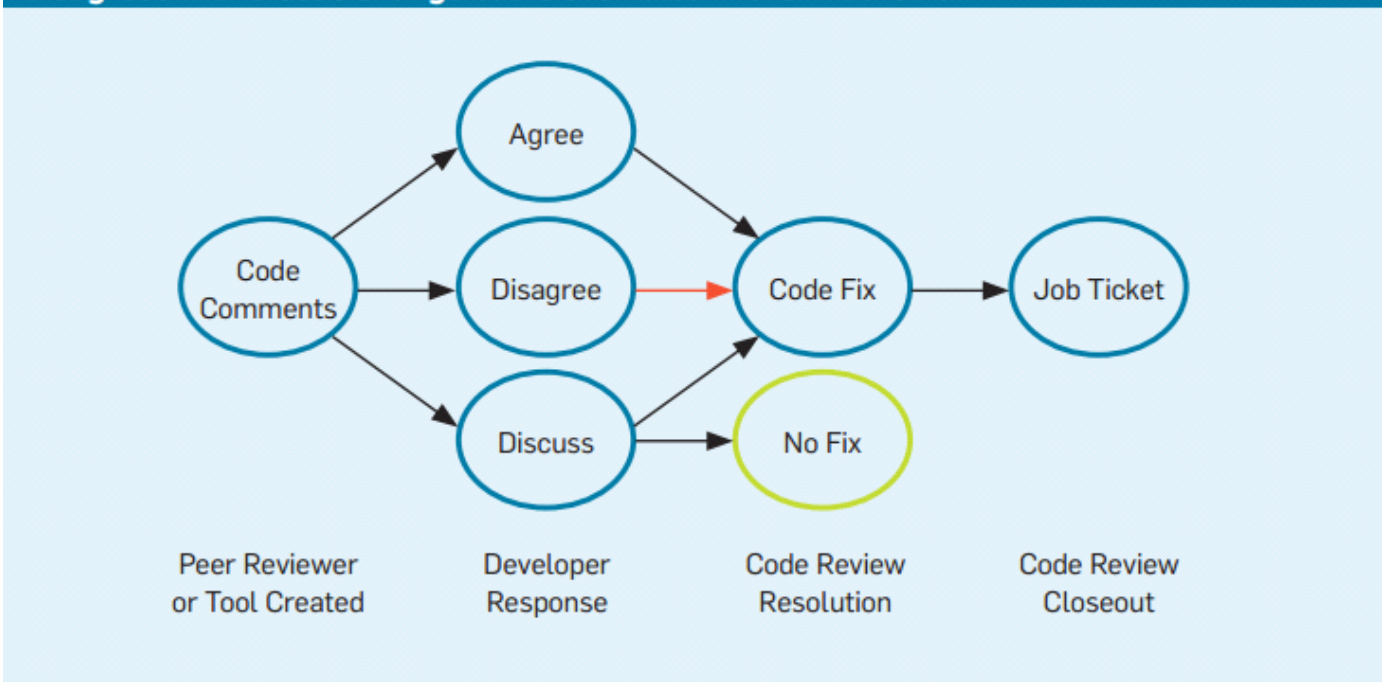
1. 基于风险的编程规则
   根据两个规则：
   a. 根据之前的经验对代码的风险评估
   b. 编程规则的协议必须被工具验证

   不同的代码有不同的重要性，因此也对不同的代码进行重要性分级



2. 基于工具的代码审阅
   就算是最强的编程规范也不能避免所有的软件缺陷。一个标准的解决方式就是同行评阅，但是对于代码量较大的代码库，普通的同行评阅的流程不行，因此采用工具Coverity，Codesonar，Semmle和Uno来检测到可能的bug并且检查。审阅者需要他们看过的代码放在scrub工具中。代码评阅流程如下：

Figure 4. Life cycle of a code comment; orange arrow indicates where the developer disagrees with a code change but is overruled in the final review.



3. 模型检查
   对于多线程程序，模型检查是最好的检查方法