刘京宗 3019213043

I'm going to briefly describe an error which impress me most. The details are as follows.

One of the most famous software errors occurred in 1996, when the European Space Agency's Ariane 5 rocket veered off course and self-destructed just 40 seconds after liftoff, resulting in a loss of over $500 million. The cause of the error was traced back to a software bug in the rocket's inertial guidance system, which was inherited from the previous version of the rocket, Ariane 4.

The bug occurred due to an integer overflow in the code that converted a 64-bit floating-point number into a 16-bit signed integer. The value of the number exceeded the maximum value that could be represented by a 16-bit signed integer, resulting in an exception that caused the rocket's guidance system to shut down.

To fix the error, the software developers should have reviewed and tested the code thoroughly, including the inherited code, to identify and fix any potential issues. They could have also implemented proper error-handling mechanisms to detect and handle such exceptions and prevent catastrophic failures.

This error serves as a cautionary tale for software developers to always test their code thoroughly, review inherited code, and implement proper error-handling mechanisms to prevent costly errors and failures.