从阿丽亚娜 501 首飞失败的软件错误所想到的

孟章荣（研究员）
（航天总公司二院 204 所　邮编 100854）

文摘　阿丽亚娜 501 发射失败引起国际宇航界的广泛关注。作者在认真研究其飞行故障调查报告的基础上，根据以往经验对其软件错误发表了自己的看法。

主题词　软件质量　软件可靠性　软件重用　操作数错误　阿丽亚娜火箭

阿丽亚娜 501 首飞失败，真是一个爆炸性新闻。全世界各类报刊及各种信息传播媒介，都先后从不同深度作了报道。特别是以 J. L. LION 教授为首的委员会，对阿丽亚娜 501 飞行故障进行了较为深入的调查、分析，非常科学、客观地写出了故障调查报告，从中我们可以得到许多有益的启示。

一、501 首飞失败故障原因的具体说法

501 首飞失败，各种报刊发表了很多有关的文章，而对故障原因的说法与调查报告却不完全一致。不过，我觉得对故障原因应该说得比较确切、一致，不能含糊其辞，更不能从自己的专业角度想当然地说明情况。关于首飞失败的故障原因，调查报告的结论是："由于惯性参考系统（SRI）计算机的程序运行中水平偏差量（BH）超出计算机软件的限值，在浮点数转换成整数操作时产生操作数错误，引起软件异常，而计算机没对此软件异常做特殊处理，采用了平时一般使用的办法——停机，因而双倍冗余的 2 个 SRI 计算机相继故障停机，致使飞行器的制导和姿态信息丢失……，最后导致运载火箭自毁，501 首飞失败"。调查报告的结论是很清楚和具体的。调查报告在故障分析中还明确指出，涉及浮点到整数转换操作有 7 个量，其中 4 个量在源码中明显加了保护，而其余 3 个量（BH 是其中之一）没加保护。Ada 语言的特点之一（安全性措施）是在运行时要对操作数的合适性作检验。因此，当对 BH 作浮点数转换整数操作时，发现 BH 数值超出计算机软件的限值，就产生操作数错误的软件异常。另外，调查报告还强调指出，致命性的错误是对此软件异常没作特殊处理，而让 SRI 的计算机停机。我们认为，这里的操作数错误的软件异常也不同于一般计算机异常中的溢出，尽管它们都是计算机程序运行过程中发生的异常事件之一。

二、故障直接原因分析中的 3 个遗憾

调查报告科学地说明了失败的技术故障事件链及产生故障的原因，并且就有关问题从软件开发起作了回顾和说明，从深层次上分析了产生故障的原因并提出了相应的建议。调查报告解释性地指出，3 个量（其中包括 BH）一般不作保护的理由：根据推断它们或受实际限制，或具有很大的安全富余度。调查报告遗憾地指出，对 BH 量，上述不作保护

的理由就阿丽亚娜 501 来说是不成立的。而保护某些和不保护另外的变量是由合同各层上的项目参与者联合作出的。对这第一个遗憾人们不禁会问，为什么会作出这样的错误决策呢？报告中提到，不是所有的转换都受保护，因为对 SRI 计算机只设置 80%的最大工作负载目标。很明显，为使 SRI 计算机的工作量不超出已设定的最大工作负载目标，不需对所有转换量都加保护，因为通常加保护就要增加工作量。保护和不保护哪些量是由合同各层次的项目参与者联合作出的，对于阿丽亚娜 501 工程项目来说，显然是作出了错误的决策。这个遗憾的深刻教训是重大工程项目的成功与否与各层负责人的学术水平、修养、道德观念和事业心休戚相关。

箭上计算机 (OBC) 误认了从 SRI 传输来的信息是第 2 个遗憾。如果从 SRI 传输到 OBC 的信息中不仅有 OBC 执行飞行程序的参考信息，而且还有 SRI 的状态信息，那么 OBC 应该区分 SRI 送来的信息类型；OBC 正常工作时，应能正确区分 SRI 送来的信息是 OBC 执行飞行程序的正常参考信息，还是 SRI 的出错状态信息。调查报告中说到 OBC 把 SRI 的出错状态信息误认为是执行飞行程序的参考信息，这就更使人难于理解。一般像这样的实时程序，在使用从传感器采样来的数据之前都需作合理性检验，剔除不合理的异常信息。我们认为，一般错误状态信息与 SRI 正常输出的敏感数据应有较大差别，只要 OBC 对 SRI 传来的敏感数据或信息进行区分和检验，就可以较早、及时地发现 SRI 的异常情况，从而可提早采取相应措施，避免事故发生。

正像调查报告指出的，致命性的错误是对 BH 浮点转换整数操作时出现的软件异常没做特殊处理。这第 3 个遗憾告诉我们，对这样关键性的实时程序，确保程序连续运行是至关重要的，决不能像平时非实时运行时那样，一出问题就轻易停机检查。在这次飞行中就因为不作特殊处理，使冗余的 2 台 SRI 计算机相继停机，致使 OBC 不能获得正确的制导和姿控信息，导致飞行失败。

三、大力加强软件工程化的某些方面

这次飞行失败使人们对软件有了一个更为清醒的认识。像这样庞大、复杂的飞行实时软件，开发不容易，检验、评审尤其复杂。委员会在调查报告中指出，软件是一种精细设计的产物，不会发生机械系统同样意义的故障。对软件的特殊性必须有充分的认识、理解。下面谈谈自己的看法。

1. 确保关键软件的质量。目前关于软件质量存在着各种各样的质量观和标准体系。对于实时系统软件归结为产品质量、关联系统的质量和工程质量等。就这 3 种质量观定义软件生存周期不同阶段判别软件质量的标准，即不同阶段确立软件的质量，在开发阶段确定工程质量；在通常软件验收、评审时使用软件的各种属性来评判、分析产品的质量；而当产品在实际系统使用时得出其关联于系统的质量。为全面获得软件质量必须综合考虑这些质量观。一般说来，人们都想尽可能地从产品质量和关联质量方面得到满意的结果，但由于实时系统软件是动态、复杂的，很难做得十全十美。只有在各开发阶段，情况相对简单时步步把关，才能保证质量。因此，对实时软件，特别强调要从工程质量上下功夫。综合这 3 种质量观就是综合软件产品在设计、制造、检验、使用和维护等各个阶段对质量的要求，它符合 ISO9000 全面质量观，即产品全生存周期的质量观。SRI 的软件在阿丽亚娜 4 中得到成功使用，即此软件关联于阿丽亚娜 4 系统的关联质量是满意的，但关联于阿丽亚娜 501，由于没有相应的维护、修改而不能通过，以失败告终。

2. 软件重用。象阿丽亚娜这样的航天系统通常都要通过各种设计、试验阶段才能达到可以使用的程度，而实际上许多软件功能模块都是重用的。软件重用不仅节省大量人

力、物力和财力，而且经过重复使用、检验、修正，相对成熟、安全、可靠，故通常认为软件重用是提高系统整体可靠性的一个有效手段。不过航天软件通常规模大，复杂多变，重用模块多，产生不匹配新系统的可能性会更大；各重用模块常常是由不同阶段、不同人员针对不同系统开发的，对于开发和检验，事先很难作出统一的规划。阿丽亚娜 5 使用的是为阿丽亚娜 4 设计的软件，在阿丽亚娜 4 研制过程中针对阿丽亚娜 4 的情况进行测试、检验。尽管阿丽亚娜 5 初始阶段情况与阿丽亚娜 4 基本一致，多余功能用不到，故没对阿丽亚娜 5 的飞行情况仔细检验、测试。从发现故障来看恰恰就是这重用模块软件不匹配于新系统，而导致软件出现异常使 SRI 计算机停机。由此可得出的深刻教训是，在重用软件时一定要检验其在新系统中使用是否匹配的问题。重用软件的输入、输出量值范围、类型通常应是在新情况下首先测试、检验的内容。对于实时程序，时间因素特别重要，实时软件运行时间受运行时各种因素影响不可能完全一样，因此常要检验关键时间点稍有变化对系统的影响。从出现故障的时间来看，实际上也已接近 SRI 这段程序运行的边界，如果在边界时刻对此程序进行测试、检验，就会发现问题。当发现不匹配的重用软件时，可使用一些特殊技巧使其协调一致，以适应新的应用情况。

3. 软件容错。在阿丽亚娜 501 的研制设计中，为提高可靠性，飞行控制系统中有很多设备都是双备份的，但实际上并没有起到应有的容错功能。首先，阿丽亚娜 501 有一个基本的主旋律：考虑硬件产生的随机故障或临时的外界干扰，而没有考虑出现软件故障的特性。通常硬件容错采用的技术是设法解决由物理衰退所引起的一些预料的故障，此技术对软件故障是不合适的。有时软件出现故障的情况很难预想，尤其是实时软件。近几年，才有人提出一些适合解决软件故障的方法。软件容错一般可分为错误探测、损害估计、错误恢复、继续服务等四个阶段。软件故障通常在输出中才表现出来，如果是设计故障，其后果很难预先估计。在实时系统中首先要考虑连续运行和超时等约束，除了那些事先预想的情况以外，损害估计和错误恢复都简化处理，重点保证继续服务。从阿丽亚娜 501 的情况看，只要不是使 SRI 的计算机停机，有用的制导和姿控信息保持不丢失，就不会发生首飞失败的惨痛结局。因此，调查报告强调指出，对此软件异常没作特殊处理是致命性的错误。

4. 计算机系统工作量富余程度。调查报告特别提到，因为 SRI 计算机承担最大工作量的 80%，不是所有的变量都加保护，这里就有如何折衷考虑为计算机工作量留有余地的问题，决不能为留有一定余量而有碍于确保程序可靠运行。实际上 Ada 语言正是为这样一些关键软件设计而研制的高级语言，它极大地关心程序运行时的安全性，因而在程序运行时要对其操作数检验是否符合程序中的定义。检验机制具体可通过 3 种方式进行：相应操作系统部分监管执行；由 Ada 编译程序在对 Ada 程序编译时形成的部分监管程序执行；在 Ada 源码程序中由编程人员编入的保护程序码。在调查报告中虽没有明确指明如何加保护，但字里行间似乎表明用第 3 种方式加的保护，这就要增加程序，加大工作量。航天软件显然是关键性软件，应用 Ada 语言编程，为了减少计算机工作量，而丢弃发挥 Ada 语言的优点是违反常理的做法。同时，Ada 程序由于这样一些安全特点在各种不同环境下运行的时间存在很大差异，并增加不少特殊的处理，较难正确估算其计算工作量，所以一般余量都留 30%以上。对于航天这样关键的系统，通常首先考虑性能，而非通常的性能价格比。在计算机性能高速发展的今天，将计算机工作量大小作为编程时考虑的约束因素真使人难以理解。

5. 软件研制的薄弱环节。调查报告在最后总结故障原因时指出，起飞 30 秒后制导和姿控信息完全丢失，信息丢失是由于 SRI 软件中技术要求和设计错误引起的。这里明确说明最直接的基本原因是 SRI 软件需求、设计的错误，这是目前软件研制中的最薄弱环节。软件系统设计人员对物理应用系统不了解，而负责物理系统的人员对计算机软件不熟悉，常常造成软件需求说明不清楚或不正确，尤其是动态、复杂的实时系统格外突出。这薄弱、困难的研制环节一直是影响软件质量的主要因素，常常导致许多软件花费很大的人力、物力研制，而最后却不能直接应用到实际系统中。当前我国许多大系统集成的软件研制中非常缺乏统一的规划、协调和管理，经常是各有各的目标，似乎各自完成了自己的目标，就算完成了任务，而最后统一的目标——真正用到实际系统中却缺少统一协调和管理。为了达到统一的目标，必须强化不同层次的项目合作者之间的技术协调。每次调度协调会都应有具体内容，接触问题的本质，决不能只负责召集开会，各管各的汇报没有协调；不仅要有定性，还应有定量的结果。阿丽亚娜 501 故障直接原因表现在软件上，而真正问题反映在整个系统集成的各部分之间的相互匹配或联接性上。

关键的软件需求设计的正确性必须通过所有各个层次的评审，让涉及所有参与工程的各方（包括外部专家）参加；同时严格进行各级的综合及系统集成的认证试验。验收测试不仅要检验接口，还要把系统作为完成特定用途的一个整体来检验。只有通过严格试验才能发现集成系统软件需求设计中存在的错误和构件之间的不匹配等。另外在实时系统研制过程中，广泛使用快速原型语言将有助于实时软件需求设计中的问题更快、更容易发现和修正。集成测试中，一般发现的大部分错误源于软件需求设计中的错误或理解。这就更进一步说明加强软件需求设计的重要性。

四、一个值得学习的题材

阿丽亚娜 501 由于软件问题首飞失败，给人们敲起了警钟，使人们对软件出错是何种性质的问题及其危害度有了现实的感悟和启示。由欧空局组织的独立调查委员会对此次事故原因调查分析后写出的故障调查报告，从技术故障事件链到故障现象说明、测试和鉴定程序，最后还提出了应得出的结论和建议。从故障调查报告中可感悟到独立调查委员会的专家们对故障的这种实事求是的科学态度，也是我们一个很好的学习题材。

(1997 年 4 月 10 日收稿)

祖国地名谜

1. 中华腾飞。(打江西一地名)
2. 振兴中华。(打江西一地名)
3. 和平城市。(打江西一地名)
4. 选招能人。(打江西一地名)
5. 永久太平。(打福建一地名)
6. 国泰民安。(打福建一地名)
7. 太平盛世。(打江苏一地名)
8. 四季丰收。(打江苏一地名)
9. 长期稳定。(打宁夏一地名)
10. 百姓富裕。(打新疆一地名)
11. 捷报传来。(打山西一地名)
12. 为官不腐。(打浙江一地名)

殷明　(谜底在本期寻找)