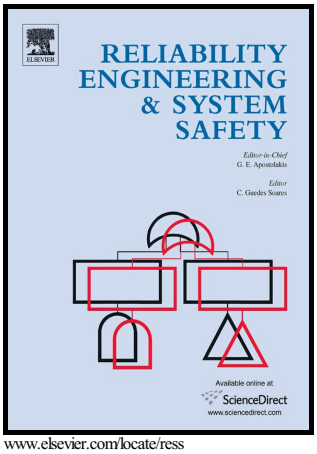
A Field Study on Root Cause Analysis of Defects in Space Software

Nuno Silva, João Carlos Cunha, Marco Vieira



www.elsevier.com/locate/ress

PII: S0951-8320(16)30403-3
DOI: http://dx.doi.org/10.1016/j.ress.2016.08.016
Reference: RESS5627

To appear in: *Reliability Engineering and System Safety*

Received date: 25 January 2016
Revised date: 10 June 2016
Accepted date: 22 August 2016

Cite this article as: Nuno Silva, João Carlos Cunha and Marco Vieira, A Field Study on Root Cause Analysis of Defects in Space Software, *Reliability Engineering and System Safety,* http://dx.doi.org/10.1016/j.ress.2016.08.016

A Field Study on Root Cause Analysis of Defects in Space Software

Nuno Silva[ab*], João Carlos Cunha[ac], Marco Vieira[a]

[a]DEI – CISUC, University of Coimbra, Polo 2, 3030-290 Coimbra, Portugal
[b]CRITICAL Software, SA, Parque Industrial de Taveiro, Lote 49, 3045-504 Coimbra, Portugal
[c]Instituto Politécnico de Coimbra – ISEC, DEIS, R. Pedro Nunes, 3030-199 Coimbra, Portugal

*Corresponding author. Tel.: +351 932 574 030; fax: +351 239 989 100, E-mail address: npsilva@dei.uc.pt.

Abstract

Critical systems, such as space systems, are developed under strict requirements envisaging high integrity in accordance to specific standards. For such software systems, an independent assessment is put into effect (Independent Software Verification and Validation - ISVV) after the regular development lifecycle and V&V activities, aiming at finding residual faults and raising confidence in the software. However, it has been observed that there is still a significant number of defects remaining at this stage, questioning the effectiveness of the previous engineering processes. This paper presents a root cause analysis of 1070 defects found in four space software projects during ISVV, by applying an improved Orthogonal Defect Classification (ODC) taxonomy and examining the defect types, triggers and impacts, in order to identify why they reached such a later stage in the development. The paper also puts forward proposals for modifications to both the software development (to prevent defects) and the V&V activities (to better detect defects) and an assessment methodology for future works on root cause analysis.

Keywords: ODC; critical systems; defect; classification; root cause analysis; dependability.

1. Introduction

Following standards and applying good engineering practices during software development is not enough to guarantee defects free software, thus additional processes, such as Independent Software Verification and Validation (ISVV), are required in critical projects. The objective of ISVV is to provide complementary and independent assessments of the software artefacts in order to find residual defects and allow their correction in a timely manner. Independence is the most important concept of ISVV and it has been referred and used in safety-critical domains such as civil aviation (DO-178B [1]), railway signalling systems (CENELEC [2]), and space (European Cooperation for Space Standardization (ECSS) working groups (e.g. [3] and [4])). However, such systems are still far from being perfect and it is common to hear about software bugs in aeronautics, car accidents caused by software problems, satellite systems that need to be patched after launch, and so on.

Previous studies have analysed the results of ISVV activities [5][6][7], looked into consolidated ISVV metrics [8] and studied the importance of independent test verification [9], showing that existing standards and good engineering practices are not enough to guarantee the required levels of safety and dependability of Critical Systems (CS). Independence of V&V avoids author bias and is often more effective at finding defects and failures. It can be managerial, financial and technical, it brings separation of concerns, complementarity, second/alternative opinions, and it also has the merit of pushing development and in-house V&V teams to focus on the quality of their work. The role of independence at early development phases is highlighted in [10] and clearly stated in the requirements of several standards such as CENELEC [2] (depending on the SIL level),