Chapter 3
Requirement-Based Test Approach and Traceability for High-Integrity Airborne Embedded Systems

Sudha Srinivasan
Aeronautical Development Agency (ADA), Bangalore, India

D. S. Chauhan
GLA University Mathura, Mathura, India

ABSTRACT

One of the biggest challenges in the development of airborne embedded systems is to ensure that the aircraft subsystem meets all its user specifications and ascertain that no important functionality is missing which leads to development of an incorrect product. This chapter proposes a methodology for achieving requirement traceability and thereby performing requirement-based testing for efficient test and evaluation of aircraft subsystems. This methodology integrates requirement traceability throughout the software development life cycle along with requirement-based testing for high-integrity software systems. The methodology has been found to be most effective in revealing errors and optimizes testing by preventing repetition of test cases across test platforms. This unique contribution has the potential to revolutionize the research world in software engineering.

DOI: 10.4018/978-1-5225-9659-2.ch003

INTRODUCTION

A well-defined software development process is essential for realization of highly reliable and safe software with cost and utilization benefits during the entire life cycle.

The software used in airborne embedded systems runs to several million lines of code and complete testing of this humongous software is a significant challenge. The process essential for certifying this airborne embedded software involves a number of software lifecycle documents where requirement traceability is essential for the entire cycle starting from capturing of system requirements to testing.

Research has shown that requirement traceability is an important contributing factor to software project failures and budget overruns. Requirements traceability refers to the ability to describe and follow the life of a requirement, in both forward and backward direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through periods of on-going refinement and iteration in any of these phases) (Gotel & Finkelstein, 1994).

Traceability makes it easy to determine what requirements, design, code, and test cases need to be updated to fulfil a change request made during the software project's development and maintenance phase and also to analyze the impact of a requirement change. Traceability links among the software development life cycle artefacts brings out how a software system was implemented to accommodate its requirements.

Many standards have been adopted in aircraft subsystems for software development like U.S.Department of Defense (DoD) standard 2167A (U.S. Department of Defense, 1988) which mandates requirement traceability.

In the conventional approach to testing, the traceability of requirements to test cases is carried out only during system testing which occurs late in the development life cycle and the resulting observations lead to large amount of rework. Thus making testing the costliest method of finding bugs.

Also, during the development of complex high integrity airborne embedded systems, there is frequent change in requirements leading to rework in design, code and testing leading to large project delays and cost implications.

Requirement based testing is a solution to these problems identified and is the suggested approach that focuses on integrating requirements with testing throughout the software development life cycle and avoids repetition of test cases across life cycle phases.

The strategy of requirement based testing is emphasized in the DO-178B guideline adopted by the aerospace industry.

In this paper, we first elaborate the observed problems in the current approach of requirement traceability and software system testing and then propose a methodology to address the challenges. The methodology optimizes software testing which