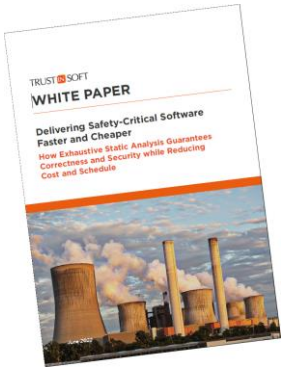
CLIENT	**TrustInSoft**
PROJECT	**White Paper on reducing cost & schedule of safety-critical software**
OBJECTIVE	**Create a white paper and related content on reducing development costs and cybersecurity vulnerabilities in industries reliant on safety/reliability-critical software.**

COPY EXCERPT

Delivering Safety-Critical Software Faster and Cheaper

How Exhaustive Static Analysis Guarantees Correctness and Security while Reducing Cost and Schedule



Call or write CopyEngineer to receive a PDF of the complete case study.

Or view/download it online at: https://bit.ly/TrustInSoft-WP-CriticalSW

Critical software has hit the "affordability wall"

Ever since General Dynamics rolled out the first fully authoritative fly-by-wire digital flight control system in the F-16 in 1978[1], critical systems have grown increasingly reliant on software. Over the last several decades, the growing size and complexity of that critical software have caused its development and verification costs to grow exponentially.

Software is now the leading cost in critical systems. According to some experts, one-third of new airplane costs are attributable to software and software development. Meanwhile, in the automotive sector, electronics and software account for 25% of the capital costs of new vehicles[2].

According to some researchers, software costs in the aerospace industry have leaped the "affordability wall." This is illustrated in Figure 1, below, where onboard software growth has been plotted logarithmically over time[3].

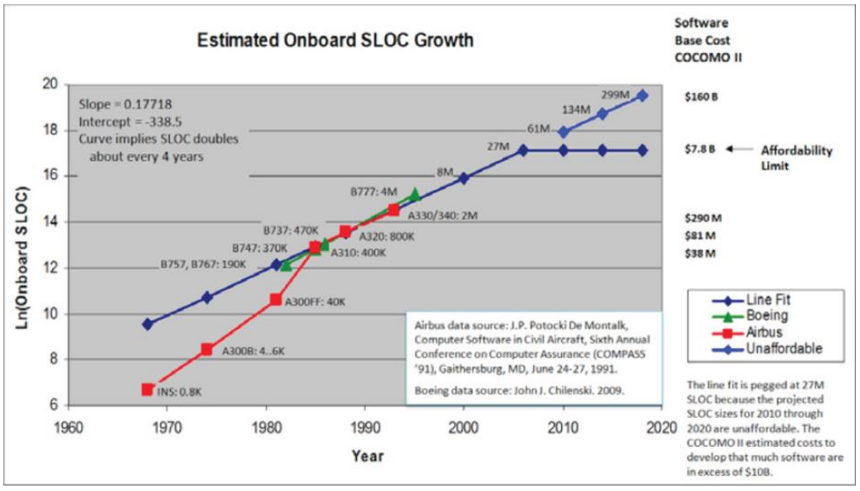


Figure 1: Estimated onboard SLOC growth in aircraft from 1960 to 2020

Source: Redman et. al, "Virtual Integration for Improved System Design," SEI, 2010.

Figure 1, created in 2010, shows that onboard software hit the affordability limit sometime around 2006, at 27 million SLOC. Since then, onboard software has continued to grow. Naturally, its complexity and cost have done likewise.

One of the primary drivers of cost in critical software is the need to assure safety.