



AdaCore CASE STUDY

How to Increase Military Software Innovation and Longevity through Reuse

Software is a critical element in almost any military application and can be a differentiator that provides strategic and tactical advantages on the battlefield. However, it also has to meet set challenges – it has to be long-lived, reliable, and easily maintainable to match the longevity of the military platform on which it is deployed. That means it has to account for both innovation and longevity from the start, and be able to work consistently and effectively over decades without the risk of failure.

Yet, in long-lived military projects, software is often being rewritten from scratch, pushing up costs and delaying overall completion. For example, the [U.S. Government Accountability Office \(GAO\)](#) highlighted software delays as a key factor in the F-35 fighter jet modernization program [failing to meet deadlines to move to full-rate production](#). This is no outlier – 10 of the Department of Defense’s (DoD’s) 15 major IT projects [are behind schedule](#).

It’s therefore time for a new approach – one that applies lessons learned from previous military software development and focuses on military grade, open source, standards-based solutions that reuse and rejuvenate existing assets. This method will deliver the innovative software capabilities that military projects require – on time and on budget.

Time to go back to the future

Software makes up a growing proportion of the overall budget for many military projects. However, while it is essential to innovate, it is just part of the military platform. It has to be able to do a job, be verifiable, long-lived and testable – just like the carbon fiber in an aircraft wing or the steel in the armor of a tank.

Yet, while carbon fiber isn’t reformulated from scratch for every new project, software often is, ignoring the existing code base that has been developed and run over decades for similar projects or applications.

Many reasons can influence the decision to rewrite existing software applications, including an overall cost analysis based on system requirements (although this often only looks at upfront expense and not project lifetime costs), performance of the existing system, hardware obsolescence, and the strength of the specific software ecosystem (including the available tools and programming skills).

The problem is that during the planning phase of military projects, programming languages can be selected that don’t necessarily have the rigor or staying power to meet long-term military-grade needs. Instead, rewriting and reusing existing, proven code can deliver the balance of innovation, cost-effectiveness and reliability that military platforms require.

We’ve been here before – back in the 1970s, the US DoD used over 450 languages within its software, leading to concerns about support, maintenance and reliability. The result? It commissioned the creation of Ada, a new, standards-based, open-source language designed specifically for long-lived, mission-critical software engineering. Ada is designed to be easy for developers to write code and also for this code to be read and understood by development teams for decades to come.

“The Ada language is the perfect fit for our global needs as it delivers unsurpassed safety and reliability.”

— Colin McDonald, Software Performance Evaluation Manager, MBDA.

Ada is now the backbone of projects across the world. World-leading missile systems company MBDA has a long-term commitment to the Ada programming language. “Successfully developing reliable, high integrity software is critical to MBDA’s world-leading portfolio of products,” said Colin McDonald, Software Performance Evaluation Manager, MBDA. “The Ada language is the perfect fit for our global needs as it delivers unsurpassed safety and reliability.” [See the full press release.](#)

Understanding the key criteria for military software

Aside from its fundamental role in the military platform, software has to meet very specific criteria. It has to be testable and verifiable to ensure it will perform consistently in use, whatever the circumstances. It must be hardware-independent, capable of meeting modern cybersecurity threats, be certifiable against military and civil avionics standards, and have an open data model to help with testing.

Demonstrating this, Airbus Helicopters selected Ada and AdaCore's GNAT Pro tool suite to develop new software components for its VSR700 prototype project. VSR700 is Airbus Helicopters' tactical unmanned aerial system (UAS) designed to meet the demanding requirements of global navies and armies in the 21st century. Commenting on the choice, Matthieu Vatinet, Head of Embedded Software Products, Airbus Helicopters said, "We selected AdaCore and the Ada language because we believe that this technology and related tools will increase the maintainability and quality of our software and make it easier to provide some evidence for certain software certification objectives." [See the full press release.](#)

Taking an open approach

The best way to cost-effectively meet these criteria is through approaches and languages based on a combination of open standards (such as The Open Group's Future Airborne Capability Environment (FACE™) Consortium or the Modular Open Systems Approach (MOSA)) and open source. This enables access to a wide range of existing software that can be re-used or rejuvenated while ensuring supply chain security through transparency and traceability, using tools such as Software Composition Analysis (SCA). In addition, ecosystems such as GCC and LLVM provide the combination of tools and technology to power open source, removing the risks of proprietary software that can add to project costs.

Adopting open source software enables military projects to securely leverage the efforts of the wider community and re-use code rather than writing it again from scratch. Given that it was created specifically for the longevity, reliability and mission-critical nature of military projects, Ada is the perfect open source programming language to support software rejuvenation. It's time to go back to the future.

Overcoming the challenges

However, many military industry HR departments are concerned about a perceived skills gap – that they can't find enough programmers willing to learn Ada for their projects.

What's needed to attract the right talent is a different approach that emphasizes the elements that will appeal to today's programmers. Ada is an open source language that provides the opportunity to be part of a growing community that uses open source tools and enables modern, innovative, state-of-the-art software development practices such as formal methods. While Ada itself has its roots in the late 70s, it has been continually advanced through the ISO-standardisation of the language, the latest version being Ada 2022. This means programmers can use innovative software tools and processes while still relying on the language for its efficiency and rigor.

Additionally, defence contractors can widen the talent pipeline themselves by working with colleges, offering apprenticeships and investing in continual training and mentoring of developers. Given the longevity of military projects, learning Ada provides programmers with a secure job for life and the chance to constantly add to their skills.

Adopting Ada for software reuse, therefore, benefits the projects, the programmers themselves and their wider communities by creating long-term, high-level employment opportunities.

Ensuring software innovation and longevity

We live in a software-driven world and it is vital to understand the critical role of software in the modern military platform. However, it doesn't need to be created afresh every time – rejuvenate existing code and focus on the right language for the job. Military projects are long-lived, complex and mission-critical. That means you need a language and an approach that fits your needs. The obvious choice is an open source language that has been deployed in numerous safety-critical platforms and is tested and proven through operational use. Adopting Ada is key to ensuring that your military software really is both innovative and immortal and innovative.

Find out how AdaCore can help you reuse and rejuvenate your existing software investments and assets. Contact sales@adacore.com