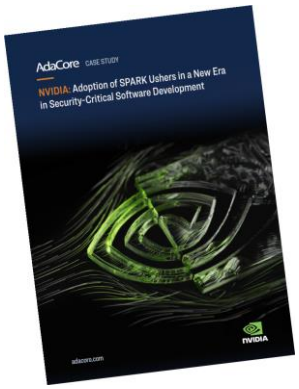
CLIENT AdaCore

PROJECT Case Study: NVIDIA

OBJECTIVE Describe why NVIDIA chose to transition from C/C++ to SPARK on critical firmware and software items, how it managed the transition, and how it has expanded its use of SPARK since proof of concept.

COPY EXCERPT

NVIDIA: Adoption of SPARK Ushers in a New Era in Security-Critical Software Development



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

Or view/download it online at: https://bit.ly/AdaCore-NVIDIA-case-study

NVIDIA Corporation, the world's best-known maker of graphics processing units (GPU), is also among the most trusted names in embedded systems, high-performance computing, and artificial intelligence. As a supplier of critical hardware, software, and firmware components across numerous sectors, they are recognized for tackling some of the tech industry's toughest problems.

Security is essential to NVIDIA's brand. However, with cybersecurity risks rising across the board, including in the verticals they serve, the company was facing the challenge of delivering more secure products without incurring a large increase in development time and cost.

An increasingly hostile cybersecurity environment

Several converging macro trends are currently putting pressure on the makers of embedded systems. Customers are demanding they demonstrate air-tight security in their products.

First, cyberattacks against firmware and hardware are on the rise. Hackers are targeting the lower levels of the technology stack hoping to exploit vulnerabilities that are difficult and expensive to correct. Due to the ubiquity of embedded systems, it is often very costly to update firmware and hardware in the field.

Second, the development and verification ecosystem hasn't kept pace with the scale of these attacks. Many languages and toolsets are insufficiently robust for critical embedded applications; they provide no guarantees during development that security vulnerabilities have been eliminated. For example, the C programming language is often preferred for embedded systems because it's great for writing fast and efficient code. But code developed in C is hard to get perfect and C lacks many verification capabilities.

Third, an industry-wide lack of secure code designers and software security practitioners is making project timescales problematic. These valuable resources have become extremely difficult to find. The cybersecurity professional organization (ISC)[2] recently calculated the global cybersecurity workforce shortage totaled some 2.7 million unfilled positions at the end of 2021. [1]

The convergence of these trends is causing extreme concern in safety-critical sectors such as medical devices, robotics, and automotive. OEMs are demanding a high degree of security assurance from their suppliers. And they have become very interested in the methods and techniques being used in secure environments…

[1] *(ISC)2 Cybersecurity Workforce Study, 2021,* (ISC)2, March 2022.