Artificial Intelligence

SPONSORED CONTENT | WHITE PAPER

Why software is crucial to achieving AI's full potential

As the volume of AI inference workloads continues to grow rapidly, CIOs and other IT leaders must consider whether their engineering and operations teams are bringing **new customer-driven products with AI and machine learning (ML) capabilities** to market fast enough and adding value where it really matters.

CIO

SPONSORED BY arm

Helping to meet this challenge, application developers are on the front lines of innovation and as a result, defining the future of how AI will transform our lives. Through AI inference-based workloads, application developers are differentiating services and products, leveraging insights from a plethora of audio, visual or text-based data. And they are moving beyond the more familiar fields of predictive text, preference analysis and camera and image processing to take an exciting leap into content generation, AI agents and autonomous cars and factories. This means grappling with secure ways to feed their data into publicly available AI models and adapting these to create unique and compelling customer value propositions.

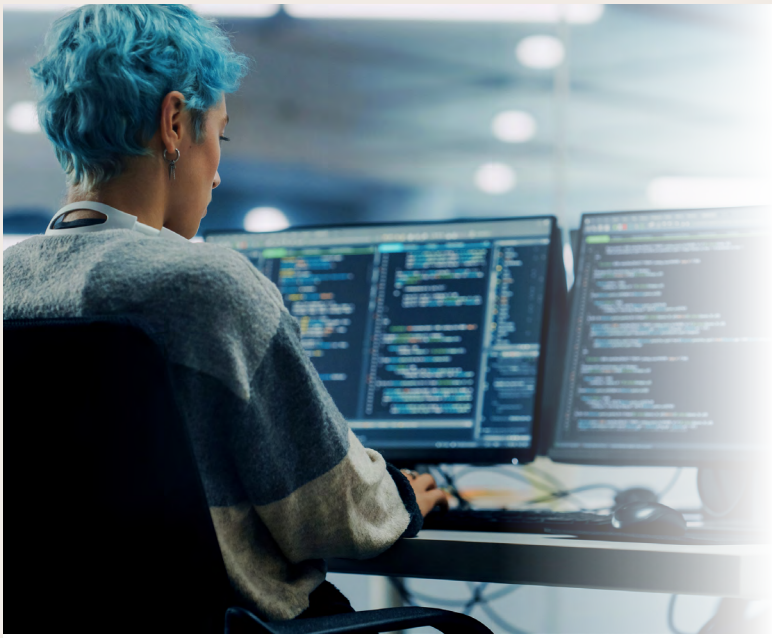
The success of new solutions presents opportunities, sending signals to the rest of the software and hardware supply chain about what's working and which opportunities should be prioritized.

However, tech leaders also need to focus on helping developers accelerate the creation and deployment of performant, secure and responsible AI applications. This may sound like a tactical preoccupation, but it's a strategic priority – particularly as AI is set to transform the way enterprises work.

AI is becoming embedded in applications everywhere, from the cloud to the edge, from consumer products to autonomous systems. Therefore, these applications are becoming essential to the operation of successful enterprises in the 21st century, creating differentiation in complex, highly competitive markets.

This is not a low-cost endeavor. Expenditure on AI in the public and private sectors worldwide is expected to rise from $235 billion today to over $630 billion by 2028, according to analyst firm IDC. It's worth noting how these funds are spent today. Hardware only accounts for 24% of these costs, whereas software accounts for 57% of the investments made this year in AI and generative AI, according to IDC.[1]

1 IDC, 'A Deep Dive into IDCs Global AI and Generative AI Spending', August 16, 2024, https://blogs.idc.com/2024/08/16/a-deep-dive-into-idcs-global-ai-and-generative-ai-spending/



> The more proprietary the ML software stack, **the more ground-up development work you have to perform.**
>
> – **Nick Horne,** VP of Machine Learning Engineering at Arm

Artificial Intelligence

As AI software development starts to account for an increasing proportion of enterprise operating costs, growing scrutiny of how software gets built is a certainty. IT leaders will no doubt be examining the performance of their development and implementation teams in forensic detail.

The Current Reality of AI Development

Today, the state of AI-based application development leaves a lot to be desired. Developers must act quickly to get to market first, adapting and scaling applications to seize new opportunities. However, developer workflows are often fragmented.

"The process contains many points of friction that require software engineers to put in extra effort to get the best out of specific hardware," says James Greenhalgh, Director of Software Product Management for Arm Neoverse (processors for cloud and high-performance computing). "You might need to take on specialist binaries which increases the complexity of your stack and perhaps ties you to one vendor. Or you run into limitations with the range of models that specialized platforms can execute."

In these situations, developers are forced to spend a lot of time on labor-intensive tasks, such as porting workloads from one set of hardware to another. They also spend less time adding value and creating points of differentiation.

"The more proprietary the ML software stack, the more ground-up development work you have to perform," says Nick Horne, VP of Machine Learning Engineering at Arm. "All of this creates a bigger investment cycle."

Pressure is building for change across developer and commercial stakeholder communities and a move to more collaboration around lower-level aspects of the software stack. Increasingly efficient code creation can free up time and space to focus on critical parts of the software development lifecycle. For example, a considered approach to embedding security and privacy by design is non-negotiable in a sector where enterprise and end-user confidence is crucial.

As inference workloads expand, development teams also have to guarantee performance on edge devices with lifespans that can extend for many years.

“Developers must have low-friction access to these platforms to update capabilities and use cases over a long period of time,” says David Maidment, Senior Director, Secure Devices Ecosystem at Arm. “A consistent, standards-based approach to securely deploying and maintaining AI models at the edge is vital. It enables both security compliance and software interoperability, which means reusing software in various places for maximum workflow efficiency. We’re developing ways of doing all of this that draw on decades of collaboration with developers.”

There’s another reason for adopting a strategic approach to AI development workflows.

“If your development teams work with tools that take them down a very specific and bespoke route, you may end up locked into a specific provider," Horne adds. “It’s better to avoid being tied to a specific piece of hardware, cloud service provider (CSP), or software platform. Working with open-source AI frameworks with good hardware abstraction minimizes loss of flexibility.”

The Rise of the Open, Standards-Based Developer Ecosystem

One long-term structural trend promises relief for IT leaders concerned about the cost of AI projects in the enterprise.

> **A consistent, standards-based approach to securely deploying and maintaining AI models at the edge is vital**. It enables both security compliance and software interoperability.
>
> – **David Maidment**, Senior Director, Secure Devices Ecosystem at Arm

> **The need to right-size compute power for each application** is something every developer and every CIO understands.
>
> — **James Greenhalgh,** Director of Software Product Management for Arm Neoverse

Artificial Intelligence

Accelerators, such as graphics processing units (GPUs) and neural processing units (NPUs) excel at the technical tasks required by most ML and AI applications, but they are costly and create challenges in terms of application development. By contrast, central processing units (CPUs) have evolved over the past decade to the point where they can meet many AI and ML demands. For example, the Armv9 and Armv8 architectures can efficiently process a broad range of AI workloads, whether in the cloud or at the edge.

"Workloads are moving around," Horne says. "Early in the history of generative AI, you needed these enormous models that would only run in the cloud. Now you can get excellent models that provide great results running on the device in your pocket, and in some cases on CPU. The benefit for enterprises is increased flexibility in terms of where you deploy, depending on your parameters for optimization."

"The need to right-size compute power for each application is something every developer and every CIO understands," Greenhalgh adds. "Not everything has to happen on specialized systems. Inference workloads alone will scale to a point where it's just not economically viable to push everything on to the highest throughput cloud instances. So, the industry is moving toward a continuum of options for running AI compute, including general purpose cloud compute, like Neoverse-powered cloud instances, on-prem and private cloud, all the way through to a plethora of edge devices."

The resulting reduction in the volume of data transfers to and from the cloud helps suppress latency, ensure improved reliability, and reduce costs. Managing workloads at the edge also

provides important benefits in terms of privacy and security, for which solutions are already available.

The rise of evolving edge computing has major implications for the way in which AI-based applications are developed. The single most important driver is the long-established availability of approaches that are open, standards-based, and designed to accelerate developer productivity.

"A standards-based approach gives you diversity of supply, which everyone cares about. It's also about enabling developers to build and deploy quickly because they have to do less per platform work," Maidment says. "We're doing as much of that work as possible on their behalf by building partnerships with open-source projects and proprietary platforms right across the industry."

However, developer efficiency is not just an issue for one type of developer. It is a pre-requisite for AI-based innovation in applications, whether running in the cloud or at the edge. As a result, Arm is making AI development more efficient in the following ways:

1. **Arm works in partnership with the open source AI community** to create opportunities for developers to easily access features in its hardware and incorporate them into applications. Within Arm, a community of experts contributes to open source projects. Arm Kleidi is a good example. It provides the key to effortless AI acceleration on high-performance CPU platforms, including best practices and open Arm technology for integrating directly across an ecosystem of ML software providers. This is accelerating AI and generative AI without the need for additional development work. The Arm KleidiAI Libraries, one key component, is already integrated into many of the world's most popular AI frameworks, including MediaPipe from Google, Meta's PyTorch and ExecuTorch and the proprietary framework Angel from Tencent.

Developer efficiency is not just an issue for one type of developer **it is a pre-requisite for AI-based innovation in applications, whether running in the cloud or at the edge.**

2. **The proliferation of AI applications is triggering a substantial rise in compute demand and complexity.** In response, Arm continues to invest in hardware optimizations that enable developers to access more capability at greater efficiency and reduce the footprint of AI models. This creates possibilities to run more AI on consumer devices and improve user experiences.

3. **IT leaders are aware of the security implications of employees running AI applications.** Trained models are worth a lot of money, and the data processed can be subject to data protection laws. That is why Arm is heavily involved in industry-wide initiatives that ensure the confidentiality and protection of applications and their data. Arm has also introduced a hardware-based trusted execution environment (TEE), managed with open source software, as part of the Armv9-A architecture. This provides complete isolation of an application and associated data from software operating at higher privilege levels, as well as other tenants. The Arm TEE extends the isolation and encryption of client applications from the CPU to include the GPU. This provides additional defense for runtime services with minimal performance hit.

4. **Arm is committed to frictionless software development** and progresses multiple initiatives to help simplify and accelerate the deployment of low-level software and firmware that underpins more strategic work. For example, Arm works to make the re-use of modular code easier and more reliable. This includes initiatives such as Linaro OneLab, Trusted Firmware and PSA Certified, fostering collaboration and reducing friction for developers by providing blueprints for secure software deployment and support, as well as GitHub Runners for testing and deploying trained AI models most efficiently in the cloud. In the automotive industry, Arm founded SOAFEE, a broad-based consortium dedicated to enabling a standards-based framework for software reuse at scale.

Artificial Intelligence

The Right Hardware for the Best Possible AI Software Development

AI promises huge commercial gains and a next generation of customer experiences that delight. But the costs of incorporating AI into products and services can be substantial, and the challenges of delivering projects on time can be significant.

It's widely accepted that AI application development in the enterprise must become faster, easier and more efficient. This has to happen without compromising security, privacy or the user experience.

Streamlining the development process is an essential part of this effort. As AI development intersects with the established CPU developer ecosystem, proprietary approaches that constrain developer productivity will lose ground to alternatives that are open, interoperable and focused on making the developer experience as efficient as possible.

Senior technology leaders and business executives can reap the benefits of this transition by empowering engineering and operations teams to make hardware choices that are accompanied by efficient tools optimized for and compatible with different operating systems and open source frameworks and libraries. Adopting this approach allows enterprises to accelerate the building of applications that deliver high-quality customer experiences across an ever-expanding range of use cases.

Arm's technology and ecosystem help future-proof AI development by emphasizing open standards, hardware abstraction, and compatibility with evolving frameworks. This approach helps to reduce the risk of vendor lock-in, ensure scalability across diverse deployment scenarios, and enable enterprises to adapt seamlessly to future advancements in AI workloads and infrastructure.◆

CIO

SPONSORED BY arm