Technical Research Report

x64 or Arm®: Which Architecture Provides the Best Windows® Experience?

Prowess Consulting performed research and testing to determine which CPU architecture provides the best performance with rich application and device compatibility for Windows users.

Executive Summary

When it comes to running Windows®, does the underlying architecture matter? Windows is just Windows, right?

In a perfect world, that would be true. But in reality, the CPU architecture—x86/x64 (32-bit and 64-bit) or Arm®—has a profound effect on Windows capabilities, application support, and even peripheral support. End users need to choose their devices for Windows carefully in order to ensure support for the applications and workloads they rely on. And software developers need to potentially double their efforts to write applications, drivers, and application programming interfaces (APIs) that are supported on both architectures.

For end users who just need to get their work done, the consequences of choosing the wrong system can be significant.

Based on extensive testing and research on Microsoft Surface® Pro 9 devices using both Intel® x86/x64 and Arm architectures, Prowess Consulting determined that devices running Windows on x64 architecture are a better fit for most users. The Intel processor–based variants that we examined offer:

- Better performance across a wide range of industry-standard benchmarks
- Broader compatibility for applications and games
- Comprehensive support for third-party drivers, support files, and peripheral devices, including monitors, webcams, and other hardware
- Support for all-day battery life[1]
- A better user experience (UX), with less friction from slow or buggy applications

In comparison, the Surface on Arm device provides all-day battery life and 5G cellular.

Industry Landscape: A Tale of Two Architectures

Given the complexities and potential confusion for users, why are there two competing architectures for Windows? A brief look at the history of each of them can help answer that question.

Note:

The industry sometimes uses the traditional term "x86 architecture" to refer to modern systems built on 64-bit ("x64" or "x86-64") architecture, which supports both 64-bit and 32-bit applications. This paper uses "x64 architecture" to emphasize support for 64-bit applications. The Intel® processor–based Microsoft Surface® Pro 9 devices used in our testing ran 64-bit (x64) architecture.

Windows on x86/x64 Architecture

x64 architecture is used by both Intel and AMD in a wide array of laptop and desktop PCs from Microsoft and leading original equipment manufacturers (OEMs), including Lenovo, HP, Dell Technologies, Acer, ASUS, Toshiba, IBM, and many more. The x64 predecessor, the x86, has a long history dating back to the Intel® 8086 microprocessor introduced in 1978.[2] These processors supported the earliest versions of Windows, beginning in 1985.[3]

In the 38 years since its inception, Windows has become one of the most popular PC operating systems in the world. The x86 architecture has been part of that journey from the beginning, allowing a rich ecosystem of software developers, applications, and tools to flourish.

Today, Intel and AMD compete to provide x64-architecture-based devices offering performance, efficiency, and affordability in a wide variety of forms built for productivity, gaming, content creation, media consumption, and other types of workloads.

Windows on Arm Architecture

Arm architecture was originally developed in 1990 as a reduced instruction set computer (RISC) CPU.[4] In comparison to complex instruction set computer (CISC) processors, RISC is designed to handle more but simpler instructions that can be processed quickly. In the mid-1990s, a Texas Instruments® chip based on Arm led to a partnership with Nokia, which used the processor to power its new line of mobile phones. In the 2000s, Arm processors were integrated into a growing number of system-on-chip (SoC) solutions because they offered a cost-effective licensing model, along with flexibility of use. Today, that legacy continues, with Arm-based solutions powering 99% of premium smartphones (according to Arm),[5] in addition to a wide variety of tablet devices.

More recently, several OEMs have brought Arm architecture to Windows and Google™ Chrome OS™ laptops. Nearly all of these devices use either Qualcomm Snapdragon® chipsets (for example, Surface Pro devices built on Qualcomm Snapdragon–based SQ1, SQ2, or SQ3 chips) or MediaTek® Kompanio® chipsets (for example, devices from Lenovo, Samsung, Huawei, HP, or Dell Technologies). For Arm device manufacturers, the intent was to bring the efficiency benefits of these smartphone-based processors to Windows laptops.

Comparing the Two Architectures

Because Arm processors have traditionally been used primarily for handheld devices, they have only been supported by Windows since 2017. That's less than a decade of support, compared to nearly four decades of support for x86/x64, giving Intel and AMD a significant head start. x64-based systems benefit from years of code refinement and optimizations, a massive developer community, a large selection of developer tools, and wide support for popular applications, as this technical research report demonstrates.

In contrast, Arm processor–based PCs have focused more on supporting web-based applications in the cloud. As a result, standalone application support has lagged behind x64-based PCs.

The scarcity of native ARM64-based applications is also due to IT business decisions: if a software company has already invested heavily in an x64 version of its application, is it worth rewriting the app to support Arm architecture? If a small to medium-sized business (SMB) has limited funds and human resources (HR), it will likely focus efforts on the solution that works for the majority of its customers—x64 architecture, in this case—rather than catering to what is currently a niche audience.

Over the last few years, Arm processor–based devices have made some gains versus the proven x64 market, but as Prowess Consulting research found in this study, Arm processor–based devices are unable to match the backward compatibility and third-party driver and peripheral support offered by devices running Windows on x64 architecture.

Emulation Adds Cross-Platform Support—At a Cost

Microsoft has taken some steps to ease the transition to Arm processors by providing an emulation layer in Windows that lets some non-native applications run on Arm. However, our testing found that this emulation can have a significant impact on performance.

Even applications that are built from scratch to support Arm architecture might have specific components or drivers that require emulation. And as our testing shows, even fully native Arm architecture–based applications can lag in performance compared to apps designed for and running on devices built on x64-based architecture.

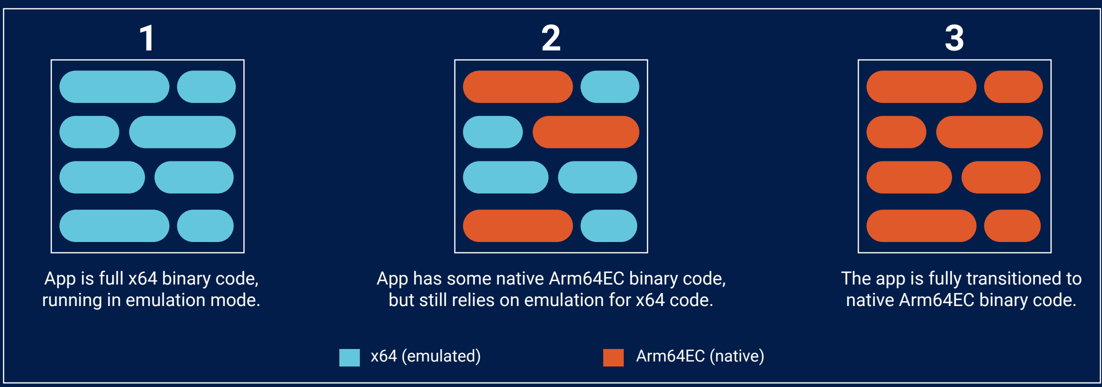
The performance gap is likely due to the Arm architecture's legacy of being built for efficiency and longer battery life in traditionally less-demanding portable devices, such as mobile phones. The architecture's original design focus might not lend itself well to the modern performance-demanding workloads required by users running laptops. For example, productivity-based multitasking workloads, such as running multiple Windows applications concurrently or using a web browser with several open tabs, did not perform as well on the Windows on Arm device we tested.

Understanding Emulation in Windows® 11 on Arm® Processors

The original ARM64 application binary interface (ABI) could only run fully native ARM64 binary applications. To ease the transition to Arm architecture for developers, Microsoft created a new ABI called Arm64EC. Microsoft describes this interface as being mostly additive to the classic ARM64 ABI, with portions added to enable x64 interoperability. According to Microsoft, Arm64EC is just as native as the original ARM64 ABI.[6]

The Arm64EC emulation layer allows developers to transition parts of the code over time, instead of completely rewriting their applications all at once. Developers can focus on transitioning the most demanding sections of code first, to make the most impact on performance.

The following graphic shows how this transition might look:



The performance of an app running in Windows on Arm architecture improves as more of the app's code is transitioned from fully emulated to fully native Arm64EC. But as our testing shows, the performance of even native Arm64 apps can lag behind apps built for Windows on Intel® architecture. (For more information, see the Performance Testing section of this paper.)

Why Arm?

Given the many challenges of supporting applications and devices on Arm architecture, why would users consider purchasing devices built for Windows on Arm? Arm promises several potential benefits for users, including:

- Power efficiency and a longer battery life (although Intel® Evo™ devices running Intel x86 architecture also promise all-day battery capabilities, defined as 9.5 or more hours on laptops with full HD displays[7])
- For Surface Pro 9 for Windows on Arm, extra webcam features and built-in 5G for constant internet access via cellular connectivity
- Instant-on capabilities (although Intel Evo devices also provide instant wake-from-sleep capabilities[8])

Given the limitations of the Arm architecture discussed earlier, is it worthwhile for users to invest in devices running Windows on Arm? To help answer this question, we conducted extensive research and performed tests on devices running Windows on both Intel x86 architecture and Arm architecture in order to compare performance and functionality.

Performance Testing: Windows on Arm Versus Windows on Intel® x64 Architecture

Our testing focused primarily on the Surface Pro 9 running on Arm architecture versus the Surface Pro 9 running on Intel architecture, as shown in Table 1. These devices are well-suited for a convenient comparison because they have similar hardware designs available in both the Intel and Arm configurations.

For performance testing, our engineers ran a series of benchmark tests comparing the following devices:

- Surface Pro 9 powered by a Microsoft® SQ3 processor (based on Arm architecture)
- Surface Pro 9 powered by an Intel® Core™ i5-1235U processor (based on Intel x64 architecture)
- Surface Pro 9 powered by an Intel Core i7-1255U processor (based on Intel x64 architecture)

For application compatibility and overall capabilities, we performed research on popular software applications to determine which ones are supported on Arm natively, through emulation, or not at all. Our testing and research examined workloads for both business and consumer users across productivity, content creation, collaboration, and gaming categories.

We also compared the same Surface devices used in performance testing to assess other differences in capabilities related to Windows features, peripheral devices, and support for third-party drivers.

Table 1 | Devices used in testing Windows® on Arm® architecture versus Windows on Intel® x64 architecture (see Appendix A for full details)

	Microsoft Surface® Pro 9 Running Windows® on Arm® Architecture	Surface Pro 9 Running Windows on Intel® x64 Architecture	
CPU	Microsoft® SQ3 processor 8 cores/8 threads (4 Arm® Cortex® X1 cores, 4 Arm Cortex A78 cores)	Intel® Core™ i5-1235U processor 10 cores/12 threads (2 Performance-cores [P-cores] + 8 Efficient-cores [E-cores])	Intel Core i7-1255U processor 10 cores/12 threads (2 P-cores + 8 E-cores)
Storage	256 GB NVM Express® (NVMe®)	256 GB NVMe	256 GB NVMe
Memory	16 GB LPDDR4x	16 GB LPDDR5	16 GB LPDDR5
Operating system (OS)	Windows 11 Home	Windows 11 Home	Windows 11 Home
BIOS	Microsoft Corporation 11.124.139	Microsoft Corporation 9.20.143	Microsoft Corporation 9.12.143

To measure and compare performance between devices, we ran the benchmarks shown in Table 2. Note that in our testing some benchmarks were supported natively by Windows on Arm architecture, some 64-bit apps ran in x64 emulation mode, and some 32-bit apps ran in x86 emulation mode.

Table 2 | Benchmark tests and emulation modes for Windows® on Arm® architecture versus Windows on Intel® x86 architecture

Benchmark Test and Applications	Windows® on Microsoft® SQ3 Processor (Arm® Architecture)	Windows on Intel® Core™ i5 and Intel Core i7 Processors (x64 Architecture)
CrossMark®		
Benchmark application	Emulation	Native
PCMark® 10 applications		
Benchmark application	Emulation	Native
Microsoft® Office applications	Mixed native/emulation*	Native
Speedometer 2.1		
Google Chrome™	Emulation	Native
Microsoft Edge®	Native	Native
WebXPRT 4		
Chrome	Emulation	Native
Edge	Native	Native
3DMark® Wildlife Extreme—Unlimited		
Benchmark application	Emulation	Native

* Although Microsoft Office applications run natively on Arm architecture, some dependent files and drivers must run in emulation mode.

Test results are shown normalized against the Surface Pro 9 powered by a Microsoft SQ3 processor based on Arm architecture, and the results are organized based on the following workload categories:

- Windows system overall performance (using CrossMark®)
- Microsoft® Office application performance (using PCMark® 10 Applications)
- Browser and web-based application performance (using Speedometer 2.1 and WebXPRT 4)
- Integrated graphics performance (using 3DMark® Wildlife Extreme Unlimited)

Windows System Overall Performance

First, our engineers ran the CrossMark benchmark to measure overall performance in Windows. As Figure 1 shows, the Intel x86 systems outperformed the SQ3 Arm architecture–based system by significant margins.

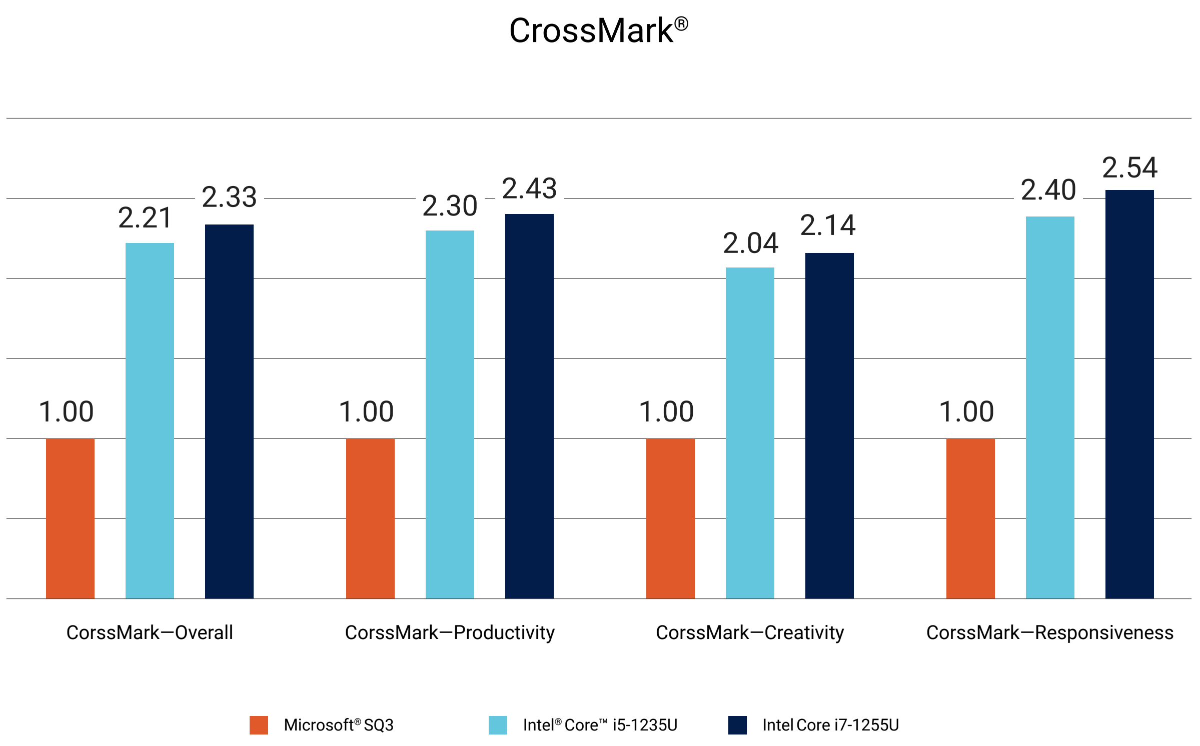


Figure 1 | CrossMark® performance

The CrossMark benchmark application itself ran in emulation mode, which might have caused the performance gaps to widen. Emulation turned out to be a common theme; our investigation and testing found fully native Arm benchmarks to be as scarce as Arm fully native general-use applications.

Microsoft® Office Application Performance

Next, our engineers ran the PCMark 10 set of benchmarks to measure application performance for Microsoft Office applications. Again, the Intel x86 systems outperformed the SQ3 Arm architecture–based system by significant margins, as shown in Figure 2.

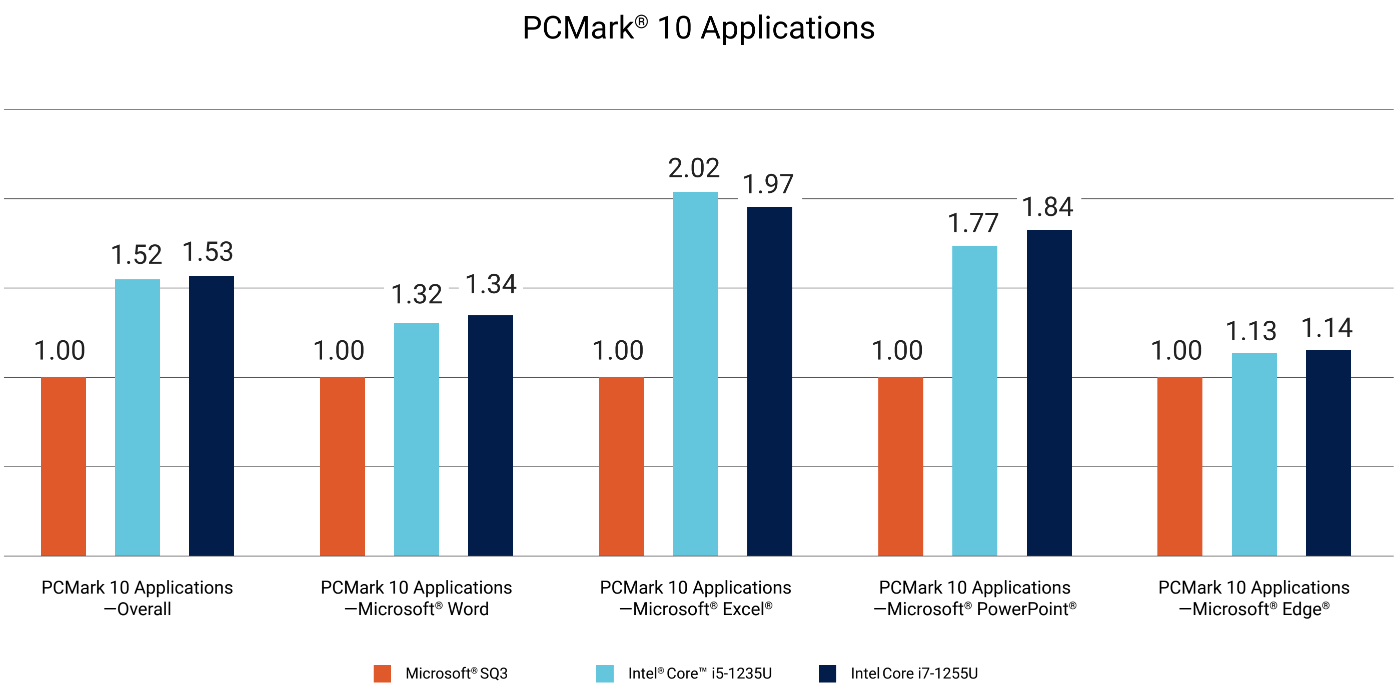


Figure 2 | PCMark® 10 application performance

In this case, the PCMark 10 benchmark ran in emulation mode, while the Microsoft Office applications were mostly native. The core applications, including Microsoft® Word, Microsoft® Excel®, Microsoft® PowerPoint®, and Microsoft Edge®, were written for native Arm architecture support, but associated libraries might be emulated, depending on the application and specific use cases.

The impact of native Arm architecture support can be seen in these results, which showed a narrower performance gap compared to the CrossMark results shown in Figure 1. However, even with a larger percentage of code native to the Arm architecture, the same applications achieved scores up to 2x greater when running on Intel x64 architecture.

Browser and Web-Based Application Performance

Next, we used the Speedometer 2.1 and WebXPRT 4 benchmark tests to examine performance for the Edge and Google Chrome™ web browsers and for browser-based applications. We expected to see a significant performance difference between the Chrome and Edge browsers running on Windows on Arm, because—unlike Chrome—Edge has been redesigned to run natively on Arm.

This discrepancy was evident in our testing, as shown in Figure 3. Both Speedometer and WebXPRT 4 benchmark tests showed significantly higher performance results for the Intel-based system when run on Chrome, compared to the Arm-based system. Speedometer showed gains of more than 4x, and WebXPRT 4 showed an advantage of more than 3x.

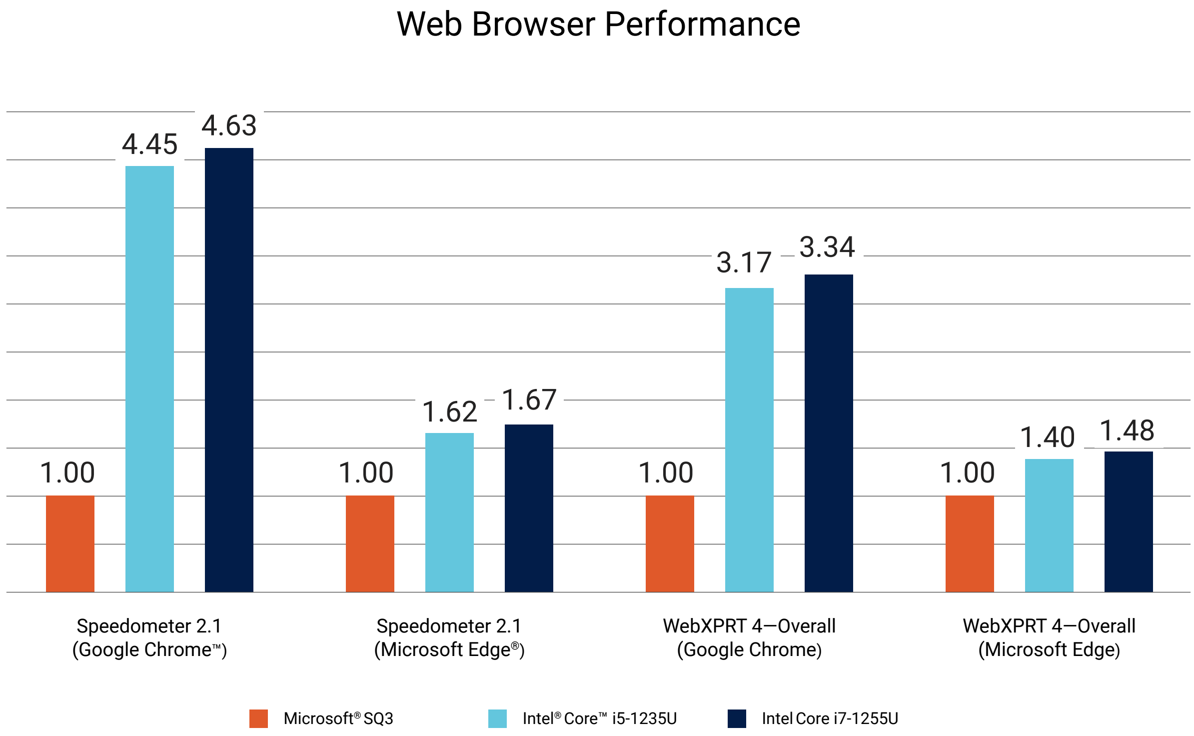


Figure 3 | Speedometer and WebXPRT 4 benchmark performance on Google Chrome™ and Microsoft Edge®

Benchmark scores were higher for the Intel-based systems even when run on Edge, which is built to run natively on Arm architecture. As seen in Figure 3, benchmark scores were 1.4x to 1.6x higher when run on Edge on the Intel x64 systems, compared to Edge on Arm architecture.

Integrated Graphics Performance

To assess performance for the integrated graphics, we used the 3DMark Wildlife Extreme Unlimited Overall benchmark test. We selected this test because it uses an API on Windows PCs to compare graphics performance on laptop computers.

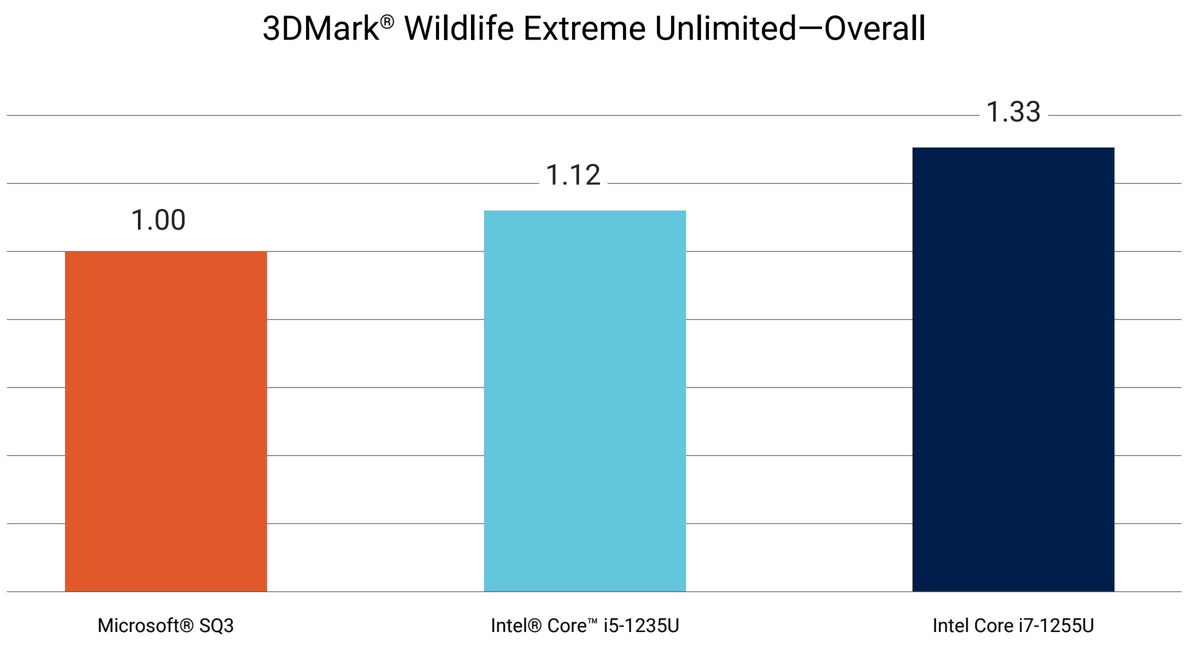


Figure 4 | 3DMark® Wildlife Extreme Unlimited Overall benchmark performance

As Figure 4 shows, graphics performance was again higher for the systems running on Intel x64 architecture, compared to the Arm architecture–based systems, with the Intel Core i7 processor–based variant showing 33% higher performance. Better scores on this benchmark can translate into higher quality renders of complex, rapidly changing scenes, resulting in a better overall viewing experience for users.

Performance Testing Summary

As our testing demonstrates, the Surface Pro 9 on Intel x64 architecture consistently beat the Surface Pro 9 running on an Arm architecture–based SQ3 processor across every category tested. From overall performance to productivity, browser performance, and integrated graphics, Windows on Intel x64 architecture showed an advantage—in some cases, beating Windows on Arm by as much as 2–4x. Even in cases where the benchmark and applications were built to run natively on Arm architecture, the Intel architecture–based systems outperformed the Arm architecture–based systems.

Based on these test results, if performance is the top consideration for choosing a device, Intel x64-based processors are demonstrably the better option. But performance isn't the only factor to consider. We also took a deeper look at other features and capabilities in order to present a more comprehensive picture.

Capabilities Analysis

We conducted extensive research to assess application, driver, and peripheral ecosystem support for devices running Windows on Arm architecture compared to devices running Windows on x64 architecture. In addition, we looked into peripherals, 5G networking, gaming, and UX.

Application Ecosystem

Nearly all applications developed for Windows are built natively for x64 architecture. Thanks to a long history of co-development between Microsoft and independent software vendors (ISVs), many such apps have been built, updated, tested, and run successfully by customers for years, resulting in highly optimized performance and an exceptional UX. In contrast, there are virtually no popular applications built for Windows on Arm architecture that are *not* already available for devices running Windows on x64 architecture.

Microsoft supports Windows developer kits built for Arm processors, but the ecosystem of native Arm architecture–based devices and applications continues to lag significantly behind the more established x86 environment. As shown in Table 3, many popular apps are unavailable with native support for Arm architecture.

Table 3 | Popular Windows® applications (business and consumer) supported natively (orange √), partially (light gray !), or not at all (dark gray X) on Intel® x64-based architecture versus Arm® architecture (as of June 2, 2023)

Application	Supported on Intel® x86/x64-Based Architecture	Supported on Arm® Architecture
Microsoft® Office	✓	✓ [9]
Microsoft Edge®	✓	✓ [10]
Microsoft Teams®	✓	✓ [11]
Microsoft® Power BI® Desktop	✓	! [12]
Google Workspace™	✓	! [13]
Google Chrome™	✓	! [14]
Google Drive™	✓	X [15]
Google Meet™	✓	X [15]
Slack®	✓	! [16]
Zoom®	✓	✓ [10]
Adobe® Reader®	✓	! [17]
Adobe® Photoshop®	✓	✓ [10]
Adobe® Lightroom®	✓	✓ [18]

To help alleviate this disadvantage, Microsoft provides an emulation layer in Windows that allows some x64-based apps to run on Windows on Arm (see the callout box, "Understanding Emulation in Windows® 11 on Arm® Processors"). However, this approach adds overhead that can have a significant performance impact, as shown in our performance testing results. Figure 1, for example, shows 2.04 to 2.54 times better performance when the CrossMark benchmark was run on x64 architecture, compared to when it was run on Arm architecture in x64 emulation mode.

Even the Microsoft Office suite of applications built for Windows on Arm is subject to performance issues related to mixed native and emulation mode. The core apps themselves are optimized for Arm, but many of the drivers, APIs, and other components in the suite are not native and must therefore make use of the emulation layer. Indeed, our testing showed up to 2x better browser performance running the PCMark 10 benchmark with Microsoft Office applications on an x64-based Surface Pro 9, compared to the Surface Pro 9 running Windows on Arm (see Figure 2).

This mixed native/emulation approach was designed to help developers slowly migrate and improve the performance of their apps, even if some of the code they rely on can't run natively yet. But as our testing showed, any degree of emulation brings a performance penalty.

To make matters more complicated, many vendors do not clearly state whether their apps are supported on Arm processors. Vendors commonly state support for macOS®, Chrome OS, or Windows, but they frequently do not provide clarity on their support for Windows on Arm—whether native or emulated. As a result, users are forced to either take a risk on their own or rely on reports that other customers have provided on websites or social-media platforms. For example, Slack® does not officially offer guidance on support for its application on Arm architecture–based systems, but several online reviews and articles (such as this CNN review of Surface Pro 9) call out its sub-par performance.[19]

Drivers, Peripherals, and Support Files

Application support is obviously a critical factor to evaluate when choosing between platforms based on x64 and Arm architecture, but drivers and other support files also need to be considered. For example, if users already own and rely on specific printers, webcams, or displays that do not have Arm architecture–supported drivers, they might be forced to purchase new peripheral devices. As Microsoft states on its website, "If a driver doesn't work, the app or hardware that relies on it won't work either (at least not fully). Peripherals and devices only work if the drivers they depend on are built into Windows 11, or if the hardware developer has released Arm64 drivers for the device."[20]

The same limitations might apply to apps that customize, enhance, or help secure the Windows experience, such as input method editors (IMEs), assistive technologies, cloud storage apps, Windows utilities, and anti-malware programs. For example, as of June 7, 2023, the Windows Fax and Scan utility is not supported on a Windows 11 PC based on Arm architecture.[20]

In each of the app examples provided above, the user would need to check with the individual software vendor to determine whether the app, utility, or peripheral is supported on Windows running on Arm architecture.

5G Cellular

Of the Surface Pro 9 models that we investigated, only the SQ3 (Arm architecture–based) variant comes with 5G cellular capability. Laptops based on Intel architecture also sometimes include a cellular option, but this requires an off-CPU chip solution that is typically more costly to the laptop manufacturer. As a result, a manufacturer might not choose to include cellular capabilities.

Windows on Arm devices, particularly those powered by Snapdragon 8cx Gen 3 processors like the Microsoft SQ3 chip, have mobile data capabilities built in. That makes it easier and less costly for vendors to include a cellular option.

Some Intel x64 architecture–based device manufacturers do provide built-in cellular connectivity, despite the additional costs. For example, the HP® Elite Dragonfly G3 notebook built on a 12th Generation Intel Core processor offers optional 5G wireless connectivity.

Built-in wireless connectivity clearly offers an advantage for road warriors, but this advantage can often be offset using an existing smartphone as a Wi-Fi hotspot, or by purchasing a separate mobile hotspot.

In addition, 5G networking availability is still limited, primarily to large urban settings. For users considering the Arm architecture–based variant of the Surface Pro 9 primarily for 5G, we recommend investigating whether 5G networking is widely available in the locations where they will be traveling and working.

Gaming

Surface Pro 9 devices are not generally considered gaming PCs. However, users might want to occasionally enjoy light gaming on these tablets. According to Microsoft, games that rely on a version of OpenGL® greater than 3.3 might not be supported on Windows on Arm devices.[20] Microsoft goes on to state that a similar limitation exists if a game relies on "anti-cheat" drivers that are not yet available for Windows 11 PCs based on Arm architecture.[20]

User Experience

UX represents a less quantitative yet equally important area of consideration for users weighing the purchase of an Arm architecture–based device compared to an Intel x64 architecture–based device.

In our research, we uncovered a wide range of articles, blog posts, and other social-media references to users experiencing a wide range of issues with running apps on Arm platforms, including slow startup times, crashes, and noticeable performance lag using or switching between non-native apps running in emulation mode. For example, The Verge article, "Microsoft Surface Pro 9 (SQ3) review: Windows on Arm is not ready," and the CNN article, "The Surface Pro 9 is a great 2-in-1 laptop — if you get the right model," both offer numerous examples of slow performance and other annoyances.[21,19]

Additionally—as noted earlier—many software vendors do not provide explicit information on support for Arm architecture. When we reviewed system requirements for various applications, we frequently found only "Windows" or "Windows 11," listed, along with other major platforms, such as macOS, Chrome OS, Android™, and iOS®. With the notable exceptions of Adobe and Microsoft, most vendors did not mention Windows on Arm support one way or another, leaving users in the dark on whether a vendor's applications would run natively or in emulation mode.

Similarly, users might face an uphill battle when trying to determine if their peripherals and third-party drivers will be supported on Arm-based devices.

Windows on Intel x64 Architecture: The Best Choice for Most Users

With the Surface Pro 9 available on both Intel x64 and Arm architecture, potential customers might want to compare the two variants when making purchasing decisions. After reviewing marketing highlights, the Surface Pro 9 built on Arm might be tempting to some users based on its promise of delivering exceptional battery life and 5G connectivity.

However, based on our testing and research, we found the x64-based version of Surface Pro 9 to be a better option for most users because it offers:

- Greater performance across all benchmarks and applications we tested, covering browsers, productivity, content creation, and gaming apps
- Much broader support for desktop software applications
- Greater support for third-party device drivers, support files, and peripheral devices, including many legacy devices
- All-day battery life (specified as 9.5 hours)[1]
- A lower cost for a system that is nearly comparable ("nearly" because the Intel model does not have the option of including the built-in 5G modem provided with the Arm architecture–based variant)[22]

We recommend that users carefully evaluate the applications they use and the performance they need for workloads before making a purchasing decision. Choosing a Windows on Arm device might require compromises, such as switching from Chrome to Edge, accepting slower performance from non-native applications, and potentially having to purchase a new display or other peripherals.

Because of these limitations, we recommend most users choose the Intel x64-based Surface Pro 9 for its better overall hardware and software experience than the Surface Pro 9 running on a Windows on Arm processor.

Appendix A: System Configurations

Testing performed by Prowess Consulting in May 2023. We used the following configurations in our testing:

	Microsoft Surface® Pro 9 Running Windows® on Arm® Architecture	Surface Pro 9 running Windows on Intel® x64 Architecture	
CPU	Microsoft® SQ3 processor 8 cores/8 threads (4 Arm® Cortex® X1 cores, 4 Arm Cortex A78 cores)	Intel® Core™ i5-1235U processor 10 cores/12 threads (2 P-cores + 8 E-cores)	Intel Core i7-1255U processor 10 cores/12 threads (2 P-cores + 8 E-cores)
Storage Brand	KIOXIA	SK hynix	Samsung
Storage	256 GB NVMe®	256 GB NVMe	256 GB NVMe
Memory	16 GB LPDDR4x	16 GB LPDDR5	16 GB LPDDR5

Operating System (OS)	Windows 11 Home	Windows 11 Home	Windows 11 Home
OS Version	22H2 22621.1635	22H2 22621.1635	22H2 22621.1635
Microsoft Office Version	2302 Build 16130.20394	2304 Build 16327.20248	2304 Build 16327.20248
Browser Version	Microsoft Edge®: 113.0.1774.35 Google Chrome™: 113.0.5672.127	Edge: 113.0.1774.35 Chrome: 113.0.5672.127	Edge: 113.0.1774.35 Chrome: 113.0.5672.127
Graphics	Qualcomm® Adreno® 8cx Gen 3	Intel® Iris® X^e graphics	Intel Iris X^e graphics
Graphics Driver Version	30.0.3564.4300	30.0.101.3118	30.0.101.3118
Resolution	2880 x 1920	2880 x 1920	2880 x 1920
Battery Size	47.7 watt-hours (Wh)	47.7 Wh	47.7 Wh
BIOS	Microsoft Corporation 11.124.139	Microsoft Corporation 9.20.143	Microsoft Corporation 9.12.143

Appendix B: Benchmarks Used in Testing

We used the following benchmarks in our testing:

- **Windows system**:
    - **CrossMark**: A benchmark from the BAPCo® consortium that measures overall system performance and system responsiveness using models of real-world applications.
- **Microsoft Office applications**:
    - **PCMark 10 Applications**: A benchmark that features a comprehensive set of tests that cover a wide variety of Microsoft Office productivity applications. The tests include a range of performance testing, custom run options, battery life profiles, and new storage benchmarks.
- **Browser and web-based applications**:
    - **Speedometer 2.1**: A benchmark that tests a browser's web app responsiveness by timing simulated user interactions. This benchmark simulates user actions for adding, completing, and removing to-do items using multiple examples in TodoMVC.
    - **WebXPRT 4**: A benchmark that compares the performance of almost any web-enabled device. It contains six scenarios based on HTML5 and JavaScript® scenarios created to mirror the tasks you do every day: photo enhancement, organizing an album using artificial intelligence (AI), stock-option pricing, encrypting notes and optical character recognition (OCR) scans, sales graphs, and online homework.
- **Integrated graphics**:
    - **3DMark Wildlife Extreme Unlimited**: A benchmark that uses the Vulkan® graphics API on Windows PCs to compare graphics performance on notebooks.

[1] "All-day battery life" defined as "9.5 or more hours of real-world battery life on laptops with full HD display." Source: Intel. "Overview of Intel® Evo™ Platform in Intel® Laptops." July 2023.

[2] Intel. Timeline website. Accessed July 2023.

[3] WinWorld. "Windows 1.0." Accessed July 2023.

[4] Arm Community Blogs. "A Brief History of Arm: Part 1." April 2015.

[5] Arm. "Consumer Technologies: Smartphones." Accessed July 2023.

[6] Microsoft. "Understanding Arm64EC ABI and assembly code." July 2022.

[7] Intel. "Overview of Intel® Evo™ Platform in Intel® Laptops." July 2023.

[8] "Instance wake-from-sleep" functionality defined as "Wake in an instant from sleep in less than 1 second." Source: Intel. "Overview of Intel® Evo™ Platform in Intel® Laptops." July 2023.

[9] Microsoft. "Microsoft 365 and Office Resources." Accessed July 2023.

[10] Android Authority. "What is Windows on Arm? Everything you need to know." January 2023.

[11] Microsoft. "Deploy, manage, and service ARM-based Surface devices." April 2023.

[12] Microsoft system requirements include only x86 and x64 platforms. For more information, see: Microsoft. Power BI download center webpage. Accessed July 2023. Users report emulation issues on online forums. For example, see: Microsoft Power BI Community. "Power BI R visuals do not work on windows on ARM." November 2022.

[13] Google Workspace™ is partially supported depending on individual service if accessed through the Microsoft Edge® browser. Note that Google Chrome™ and Google Drive™ are not supported on Arm® architecture. For more details, see: Google. "Chrome browser system requirements." Accessed July 2023. Also see: Google. "Service-specific Google Workspace requirements." Accessed July 2023.

[14] Google. "Chrome browser system requirements." Accessed July 2023.

[15] Google. "Service-specific Google Workspace requirements." Accessed July 2023.

[16] Slack® system requirements do not call out Arm® architecture. Online articles indicate performance issues. For more information, see: CNN. "The Surface Pro 9 is a great 2-in-1 laptop — if you get the right model." November 2022.

[17] Adobe® Reader® runs in 32-bit emulation mode on Arm® architecture. Source: Adobe. "ARM processor support | Adobe Acrobat and Acrobat Reader on Windows." May 2023.

[18] Adobe. "Will Adobe apps work on Windows computers that use ARM processors?" November 2022.

[19] CNN. "The Surface Pro 9 is a great 2-in-1 laptop — if you get the right model." November 2022.

[20] Microsoft. "Windows Arm-based PCs FAQ." Accessed July 2023.

[21] The Verge. "Microsoft Surface Pro 9 (SQ3) review: Windows on Arm is not ready." November 2022.

[22] "Lower cost" claim based on Microsoft Surface® Pro 9 devices with 8 GB RAM and 256 GB solid-state drives (SSDs). Microsoft. Microsoft Surface Build your device webpage. Accessed June 2023.

PROWESS

The analysis in this document was done by Prowess Consulting and commissioned by Intel.
Results have been simulated and are provided for informational purposes only.
Any difference in system hardware or software design or configuration may affect actual performance.

Published: 8/15/2023
0823/230108