INTEL SIMICS

Simulate Anything, Chip to System

Electronic systems continue to increase in complexity, with heterogeneous processor architectures, multilayered software stacks, and expanded network connectivity. With more connections comes additional security risk, which requires thorough testing and mitigation as early as possible in the product lifecycle.

Meanwhile, organizations are incorporating DevOps practices to meet customer and market expectations for quicker, incremental deliveries. Such methodologies rely on fast iterations for testing, feedback, and deployment. Cross-functional teams need shared tools to communicate and collaborate.

Intel® Simics® helps organizations meet these testing requirements by giving developers on-demand access to any target system at any time. It offers efficient collaboration between development and quality assurance teams, which streamlines complex tasks such as certification testing. Simics provides an automation API, so organizations can reap the business benefits of DevOps and continuous development practices. That lets companies deliver better, more secure software, faster — even for complex, embedded, connected, and large systems deployed at the intelligent edge.

DEVELOP SOFTWARE IN A VIRTUAL ENVIRONMENT

Simics provides the access, automation, and collaboration required to enable DevOps and continuous development practices. By using virtual platforms and simulation, software developers can decouple their work from physical hardware and its limitations. Access to virtual hardware allows developers to perform continuous integration and automated testing sooner in the development cycle — even before the hardware design is finalized — and to perform both testing and debugging during design and prototyping phases.

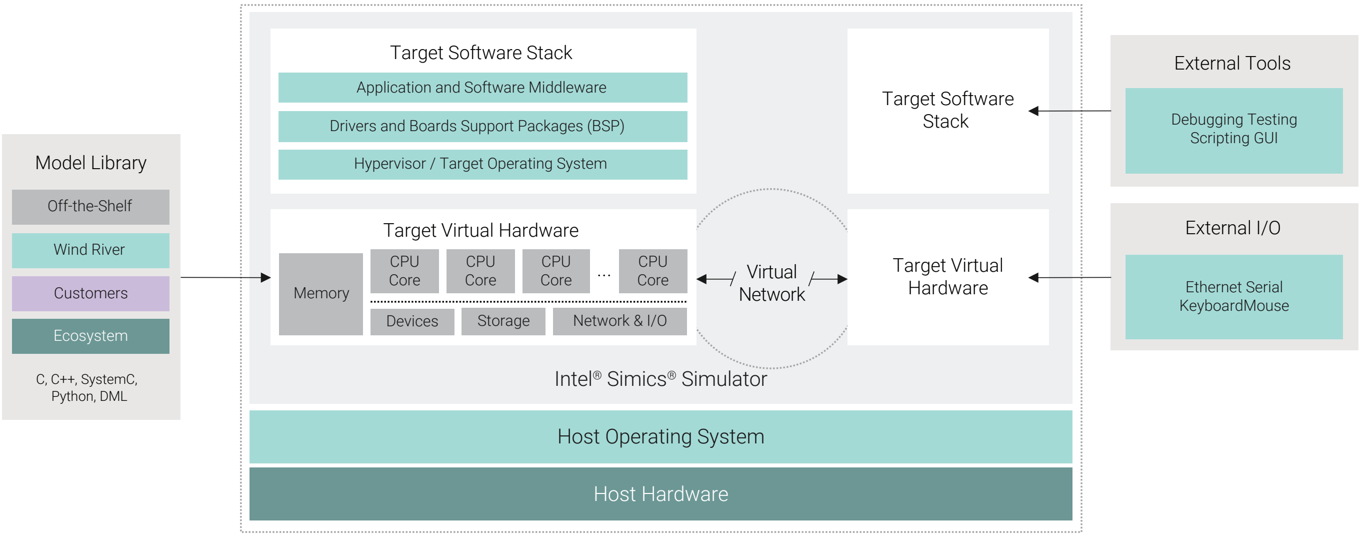

Figure 1. Intel Simics simulator: Extended architecture overview

Software developers use Simics to simulate nearly anything, from a single chip to complete systems and networks of any size and complexity. The Simics simulation of a target hardware system can run unmodified target software. It includes the same boot loader, basic input-output system (BIOS), firmware, operating system (OS), board support package (BSP), middleware, and applications as the hardware. Security threats can be tested in a safe and controlled environment by using fault injection in a simulated system.

MAXIMIZE PRODUCT DEVELOPMENT VELOCITY

Simics enables efficient practices and higher velocity for development and testing. With seamless and reliable access to the Simics virtual hardware platform, teams can refine in real time, iterate continuously, and move quickly through design, development, and testing processes to build their systems incrementally. Simics provides an environment where team members can share running systems and set up configurations with each other, rather than manually documenting and describing system setups and software behavior. Furthermore, Simics helps break the physical limitations of real-world hardware by automating and enabling scenarios that are impossible to replicate in the real world.

OPTIMIZE SYSTEM DEBUGGING

Complex and connected systems are difficult to debug and manage. While traditional development tools can help developers track down bugs related to a single board or software process, finding a bug in a system of many boards and processor cores is a daunting task. For example, if a developer stops one process or thread with a traditional debugger, other parts in the system continue to execute, making it impossible to get a globally coherent view of the target system state.

Simics provides access to, visibility into, and control over all boards and processor cores in a system. The whole system can be inspected and debugged as a unit. A developer or tester can create a checkpoint that captures the entire system state. This state can be passed to another developer, who can then inspect the precise hardware and software state, replay recorded executions, and continue execution as if the process had never stopped.

Simics virtual platforms are easier to configure and manage than physical hardware. With an infinite supply of each type of board, scalability testing can go beyond what is possible with physical hardware. Also, developers and testers can inject faults into any part of the system in a nondestructive and repeatable manner.

Simics expands beyond the physical limitations of hardware, with no degradation in software execution. Even if simulation breaks the physical barriers and goes into the realm of hypothetic hardware configurations, it still guarantees software execution performance as in real configurations.

ON-DEMAND VIRTUAL LABS

One obstacle in developing, debugging, integrating, and testing electronic systems is that target hardware and physical labs are not always available in an operational state for everyone, or access to hardware and physical labs are subject to long waiting times. This limited access means engineers must make do with less-than-ideal substitutes, such as reference boards or host-based development.

With Simics, you can build a virtual lab that is on demand — available for any team member, at any point in time, at any location in the world, and with any amount of hardware. The virtual lab is not just a piece of the system; it can be the complete system. This capability allows people to do their work in the context of that complete system instead of just a part of it.

For developers who support many different configurations of their target hardware, a Simics virtual platform makes it easy to manage multiple configurations of components, which is especially useful for compatibility testing and testing specific customer scenarios.

SUPPORT CONTINUOUS PRACTICES THROUGHOUT THE ENTIRE PRODUCT LIFECYCLE

Simics supports the entire product lifecycle, from design through product deployment, validation, certification, and maintenance.

Design

Explore different hardware and software design options, supported by concrete performance and reliability data, leading to more robust and optimized system architectures.

- Explore system design choices and hardware/software interfaces before silicon is locked down.
- Evolve the product. Start with the last generation and implement the next generation of functionality, component by component. Experiment with hardware and software variations and analyze impacts.

Develop

Develop, debug, and test low-level software, BSPs, and OS configurations even when hardware is not available, and without physical hardware constraints..

- Parallelize and reduce dependencies between software and hardware development.
- Provide application developers with virtual target hardware so they can test on the real system at any time.
- Use debugging features such as checkpoints, record, replay, reverse debugging, and code and hardware breakpoints.

Validate

Start testing and automation early in the development process. Do continuous hardware and software integration early on with virtual hardware, expanding to physical hardware as it becomes available.

- Build more levels of intermediate setups than are available with hardware, to facilitate continuous integration.
- Test fault tolerance with Simics fault injection. Cover corner cases that cannot be reached in hardware.
- Test, reset, and test again using nondestructive security testing, as many times as needed, without damaging the system.
- Automate and parallelize testing and expand coverage of target configurations using Simics scripting.
- Use simulation labs in addition to hardware labs to save developer time, reduce waiting time to run tests, and shorten feedback loops.
- Integrate Simics models of computer hardware with external models of the physical world or system environment for testing and integration on the entire system.
- Automate regression testing and continuous integration by tying Simics into existing workflows of software build and test.

Certification

Passing certification requires ample preparation, which can be time-consuming, stressful, and expensive. Simulation can dramatically reduce preparation effort while simultaneously increasing the likelihood of certifying — the first time.

- Streamline certification tasks by sharing the complete virtual system or parts of it in the process of generating the required artifacts.
- Use pre-built and optimized models for developers, testers, and certification professionals to collaborate in virtual environments.
- Distribute and share custom configurations easily by offering on-demand access to any target system at any time.

Maintain

Enable long-term customer support and maintenance by setting up a virtual lab and equipping geographically distributed teams with a virtual equivalent of the deployed product.

- Create virtual configurations matching customer system configurations to investigate issues and aid support.
- Maintain legacy products indefinitely without worrying about hardware obsolescence and decommissioning.

SUPPORTED TARGET ARCHITECTURES

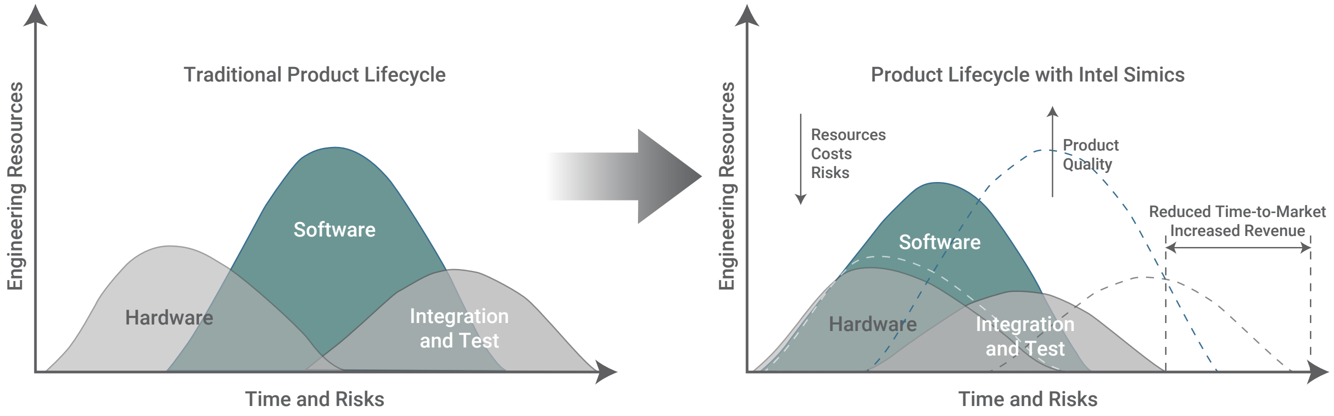

Figure 2. Simics shortens the traditional product lifecycle

- Intel® Architecture, 32-bit and 64-bit
- Arm®, 32-bit and 64-bit
- RISC-V, 32-bit and 64-bit
- PowerPC, 32-bit and 64-bit (with selected version of Simics)

KEY SIMICS CAPABILITIES

- Immediately run a simulation of popular target architectures using the built-in quick-start platforms for Arm, Intel, and RISC-V.
- Simulate target systems, including processors, devices, full boards, and systems.
- Run the same software on Simics that runs on the physical target.
- Enable everyone with access to the virtual system within teams, across teams, and with partners.
- Analyze and debug the full system as a unit, not just as a single board.
- Share system state, execution history, and virtual hardware setups with anyone, anywhere.
- Resume execution from the exact point at which it was last saved.
- Inject faults to test system robustness and test diagnostic software.
- Automate manual steps and repetitive tasks with full scripting capability.
- Integrate with existing software development environment and automated workflows, or have Wind River® Professional Services do it for your organization.
- Integrate physics and environment simulators to build complete system simulations, or let Wind River Professional Services do so.
- Use the Simics command-line interface, C/C++ source code debugger, or an organization’s standard software toolset.
- Use a prepackaged virtual platform from Wind River, build your own, or have Wind River Professional Services build one.

** Intel® and Simics® are trademarks of Intel Corporation or its subsidiaries.*

WNDRVR

Wind River is a global leader of software for the intelligent edge. Its technology has been powering the safest, most secure devices since 1981 and is in billions of products. Wind River is accelerating the digital transformation of mission-critical intelligent systems that demand the highest levels of security, safety, and reliability.