WIND™

AN INTEL COMPANY

WIND RIVER SIMICS

Electronic systems are becoming increasingly complex, with more hardware, more software, and more connectivity. Current systems are software intensive, often containing multiple heterogeneous processors running complex multilayered software stacks. They are also inevitably part of a larger system—in particular, networks and the Internet of Things. With an increasing emphasis on smart and connected systems, complexity in software and hardware is unavoidable. In addition, innovation cycles become shorter, and developers turn to agile and continuous development practices to meet customer and company expectations on quick deliveries. Such methodologies rely on fast iterations for test, feedback, and deployment. Collaborative and cross-functional teams need tools to communicate and share a common development baseline.

Wind River® Simics® allows developers to have on-demand and easy access to any target system, more efficient collaboration between developers, and more efficient and stable automation, enabling organizations to reap the business benefits of agile and continuous development practices to create and deliver better software, faster—even for complex, embedded, connected, and large IoT systems.

DEVELOP SOFTWARE IN A VIRTUAL ENVIRONMENT

Simics provides the access, automation, and collaboration required to enable agile and continuous development practices. By using virtual platforms and simulation, software developers can decouple their work from physical hardware and its limitations during development. Access to virtual hardware allows developers to do continuous integration and automated testing much sooner in the development cycle—even before the hardware design is finalized—as well as perform both testing and debugging during design and prototyping phases. All team members can have unlimited access to virtual hardware, allowing new ways of working and improved collaboration throughout the lifecycle. Virtual hardware can be shipped anywhere, at any time, improving collaboration and ensuring a common baseline for all developers. Ultimately, developers can dramatically reduce risks of shipping late, overrunning budget, and having quality problems.

Software developers use Simics to simulate nearly anything, from a single chip all the way up to complete systems and networks of any size or complexity. A Simics simulation of a target system can run unmodified target software (the same boot loader, BIOS, firmware, operating system, board support package (BSP), middleware, and applications as the hardware), which means users can reap the benefits of using a pure software tool.

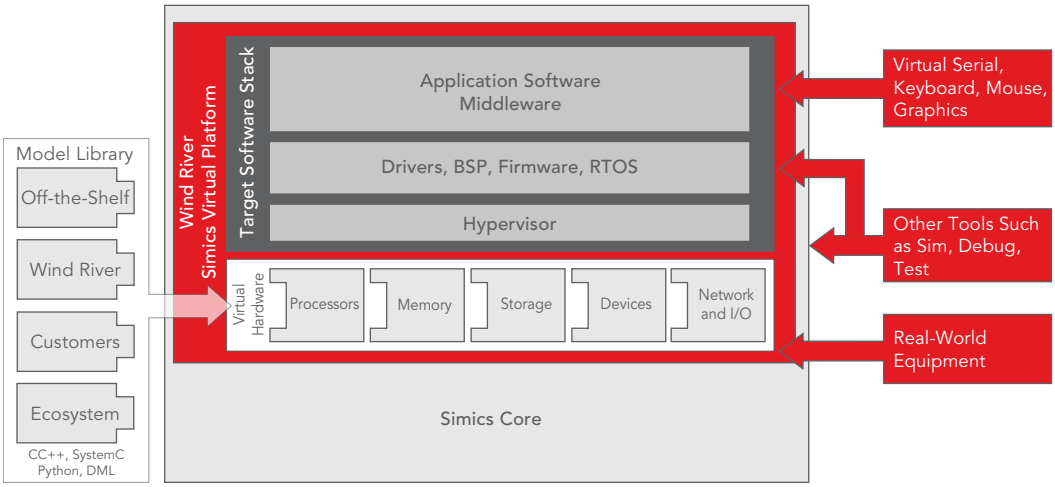

Figure 1: Simulate your entire target system

MAXIMIZE PRODUCT DEVELOPMENT VELOCITY THROUGH ACCESS, COLLABORATION, AND AUTOMATION

Simics enables more efficient development 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 through design, development, and test quickly to incrementally build their system. Simics improves team communication by providing an environment where team members can share actual running systems and system setups with each other, rather than documenting and describing system setups and software behavior. Furthermore, Simics helps shorten development and testing cycles by automating what is impossible to automate with physical hardware.

OPTIMIZE SYSTEM DEVELOPMENT

Complex and connected systems are difficult to debug and manage. While traditional development tools can help you 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 you stop one process or thread with a traditional debugger, other parts in the system will 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 the system. Single-stepping forward and in reverse applies to the system as a whole; the whole system can be inspected and debugged as a unit. Furthermore, a checkpoint—or snapshot—can be created, capturing 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 it never stopped.

Simics virtual platforms are easier to configure and manage than physical hardware. With the infinite supply of each type of board, scalability testing can go beyond what is possible with physical hardware. In addition, faults can be injected into any part of the system, in a non-destructive and repeatable manner. And Simics can simulate systems containing hundreds of processors with a performance level that enables real software target loads to execute.

ON-DEMAND VIRTUAL LABS

One of the biggest obstacles with developing, debugging, integrating, and testing an electronic system is that target hardware and physical labs are not always available for everyone, in an operational state, or they are subject to long waiting times to access. This limited access means engineers have to 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. Furthermore, the virtual lab is not just a piece of the system, it can be the complete system. This capability allows users to do their work in the context of a 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 and maintenance.

Design

- Explore system design choices and hardware/software interfaces, before silicon is locked down.
- Evolve the product over generations—start with the last generation, and implement the next generation 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 the constraints of physical hardware.
- Parallelize and reduce dependencies between software and hardware development.
- Provide all application developers with virtual target hardware to enable testing on the real system at any time.
- Leverage advanced debugging capabilities such as checkpoints, record, replay, reverse debugging, and advanced code and hardware breakpoints.

Integrate and Test

- Start testing and automation early in the development process. Do continuous hardware and software integration early, on 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.
- Automate and parallelize testing and expand coverage of target configurations using Simics scripting.
- Save developer time, reduce waiting time to run tests, and shorten feedback loops by using simulation labs in addition to hardware labs.
- Do test and integration on the entire system by integrating Simics models of computer hardware with external models of the physical world or system environment.
- Automate regression testing and continuous integration by tying Simics into existing workflows of software build and test.

Deliver

- Enable partners and ecosystems by sharing the complete virtual system, or parts of it.
- Distribute and share customer-specific configurations easily.
- Keep the ecosystem updated as the system evolves with new hardware and software.

Maintain

- Aid customer support by setting up a virtual lab and equipping geographically distributed teams with a virtual equivalent of the end 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.

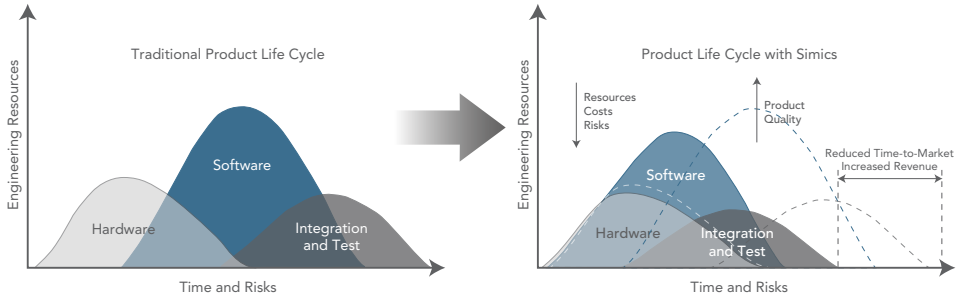

Figure 2: Wind River Simics accelerates product development, improves product quality, and reduces risks and costs

SUPPORTED TARGET ARCHITECTURES

- ARM, 32-bit and 64-bit
- Intel Architecture, 32-bit and 64-bit
- MIPS, 32-bit and 64-bit
- Power, 32-bit and 64-bit
- Others

KEY WIND RIVER SIMICS CAPABILITIES

- Immediately run a simulation of popular target architectures using the built-in Quick Start Platforms for ARM®, Intel®, and PowerPC® that come with Simics.
- Simulate your target system, 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 and your ecosystem.
- Analyze and debug the full system as a unit (not just a single board).
- Debug efficiently by running the simulation forward or in reverse.
- Share system state, execution history, and virtual hardware setups with anyone, anywhere.
- Resume execution from the exact point at which you last saved it.
- Inject faults to test system robustness and test diagnostic software.
- Automate manual steps and repetitive tasks with full scripting capability.
- Integrate with your existing software development environment and automated work flows. Or have Wind River Professional Services do it for you.
- Integrate physics and environment simulators to build complete system simulations. Or have Wind River Professional Services do it for you.
- Use the Simics Eclipse-based, C/C++ source code debugger, or use your standard software toolset.
- Use a prepackaged virtual platform from Wind River, build your own, or have Wind River Professional Services build one for you.

WIND
AN INTEL COMPANY

Wind River is a global leader in delivering software for the Internet of Things. The company's technology is found in more than 2 billion devices, backed by world-class professional services and customer support. Wind River delivers the software and expertise that enable the innovation and deployment of safe, secure, and reliable intelligent systems.