Simics: A Full System Simulation Platform

Peter S. Magnusson et al
Virtutech

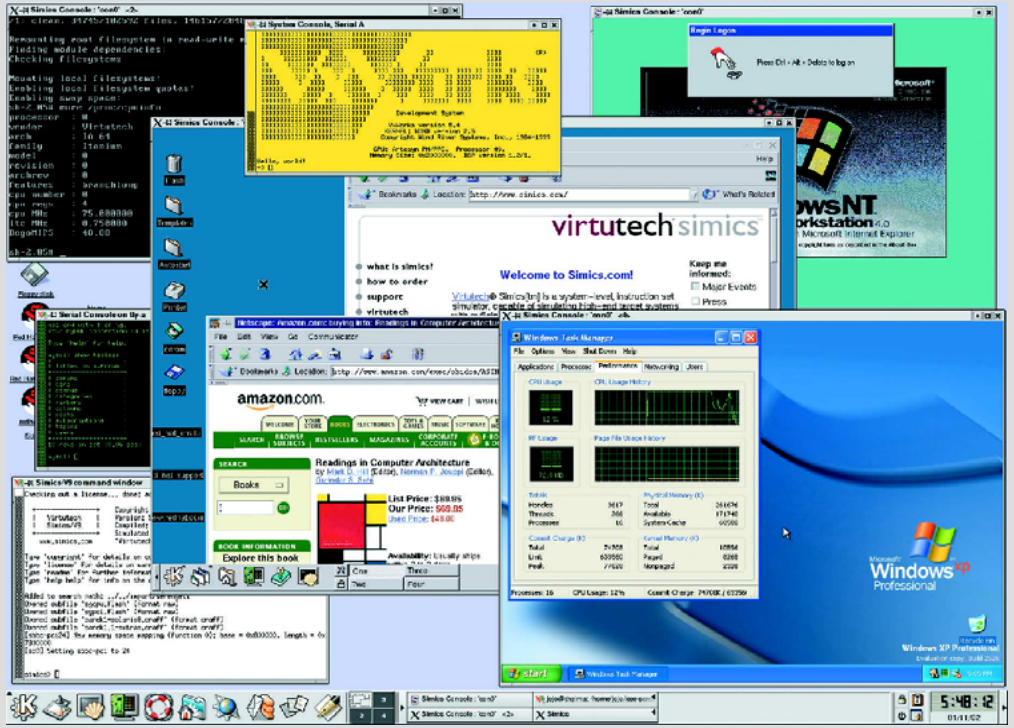
Presented by Clif Kerr

Overview

- Accurately simulate function of a given architecture
- Facilitate timing simulation with hooks to third-party timing simulators
- Operate fast enough to run real workloads

About Simics

- Functional simulation of UltraSparc, Alpha, x86 (32 bit and 64 bit extended), PowerPC, Itanium, MIPS, and ARM processors
  - Simulations run unmodified operating systems and code



Simics Applications

- Processor Design
- Multiprocessor Architecture
- Operating System Development and Emulation
- Debugging

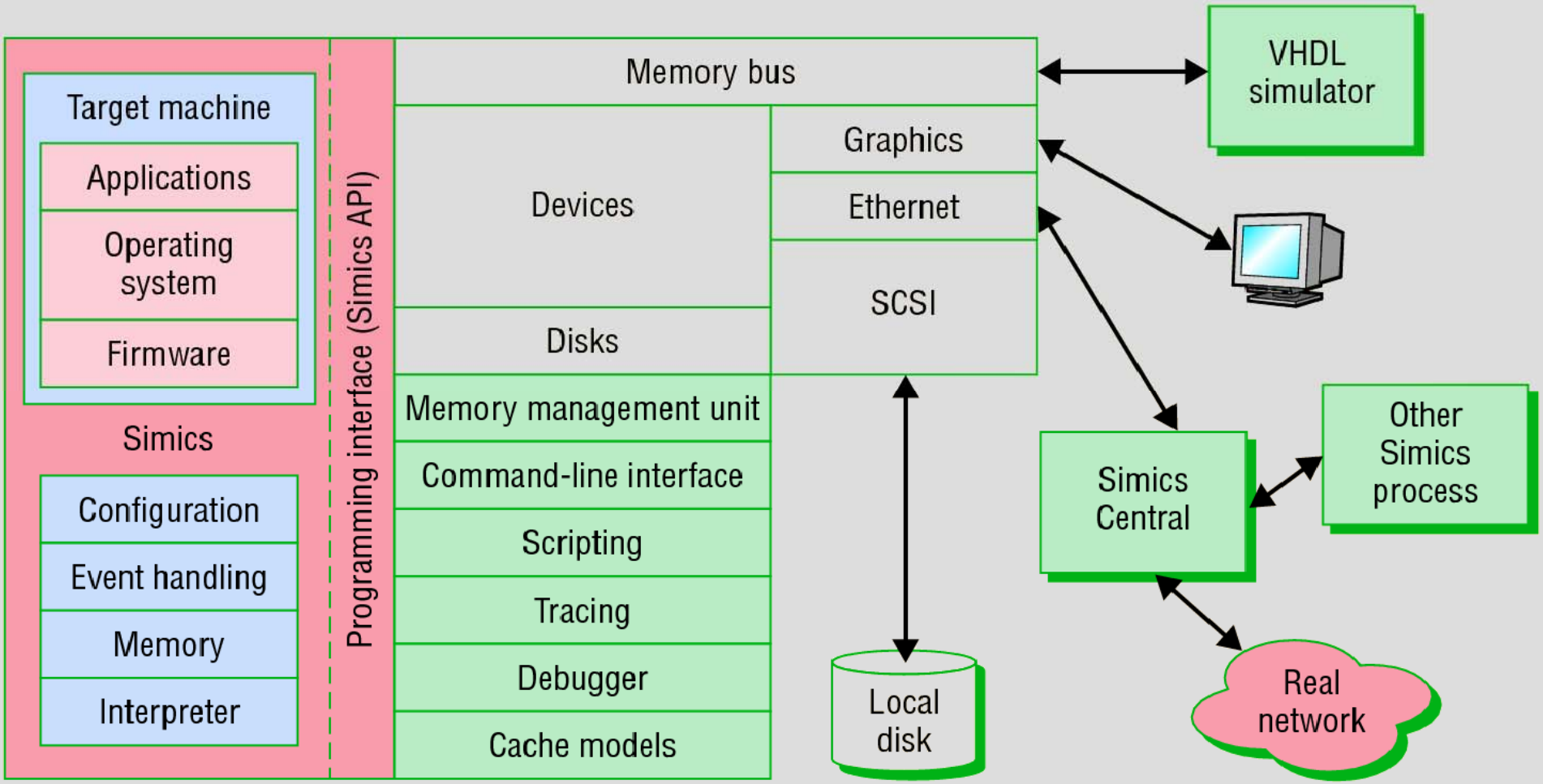
Implimentation

- Simics Central – coordinates communication between Simics modules in a simulated network
- Memory simulation through simulated transaction cache
- Interpreter generated from high-level architecture specification

Extensibility

- New device modules “plug in” to Simics framework
- Simics API provides numerous functions, data types, and interfaces defined for user-defined extensions

Simics Architecture



Performance

- Simics can simulate real workloads at reasonable speeds – on the order of a million instructions per second

Table 1. Simics performance of target systems for a variety of operating-system boot workloads.

Target	Boot workload	Instructions	Time (sec)	MIPS
Alpha-ev5	Tru64	2,112,119,247	354	5.9
Alpha-ev5	Linux	1,201,600,120	164	7.3
Sparc-u2	Solaris 8[1]	1,597,537,438	284	5.6
Sparc-u3	Solaris 8[1]	6,155,835,717	987	6.2
x86-p2	Linux[2]	1,299,639,608	227	5.7
x86-p2	Windows XP	3,129,351,000	1,518	2.1
x86-64	Linux[2]	1,299,639,608	285	4.5
Itanium	Linux	4,644,372,142	1,470	3.2
PPC-750	VxWorks	1,179,516,468	136	8.7
PPC-750	Linux[3]	498,836,969	53	9.3

Questions

- Is the reported performance a result of optimization or simulator simplicity?
- What are some of the potential drawbacks of making a completely general simulation environment?