What is Simics®?

Simics® - an adaptive simulation platform that enable customers to define, develop and deploy electronics systems more efficiently
No Impact on the Target Software

The software can’t tell the difference
Runs binaries from real target

User program
Java VM | DB | Middleware

Operating system
Drivers | Hardware abstraction layer | Boot firmware

Virtual Hardware
CPU | Bus | Disk | Network
CPU | PCI | Disk Ctrl | User Intf device
RAM | I2C | FLASH | A/D
ROM | CPU | RAM | PCI

Run your system software on your desktop
Complete production software
Identical build tools chain
Simulated (virtual) hardware
Simics Scales

Examples

» Satellite constellation, telecom network

» Telecom rack, avionics bay, blade server

» MPC8572DS board, standard PC, DSP farm

» PCIe, RapidIO, I\textsuperscript{2}C, Custom FPGA

» PPC440GP, P4080, OMAP, Octeon CN3860

» PPC750, Core 2, e500mc, POWER6, MIPS64

Copyright © 2009 Virtutech Inc. Confidential
How is Simics Used?

• System Development and Integration
• Large Systems Modeling
• Training platform
• Product Support platforms

• As a new, disruptive platform for Product Life Cycle process change:
  – System/architecture definition
  – Software development
  – Deployment
  – Integration and test
  – Demo, support and training
  – Communication between groups
• Reconfigurable Environment for Testing of Software Systems

• Dynamic simulation environment

• Capabilities
  – Create standalone simulation environment
    ▪ Simulate entire vehicle environment – including analog inputs
  – Scalable simulation – up to 100s of CPUs

• Includes Simics abilities
  – Scriptable test scenarios
  – Automated
    ▪ Batch mode testing
  – Post-processed results
REALSS Contains

- **Connect to real subsystems**
- **Simulation of hardware**
- **Runs production binary**
- **Virtutech Simics®**

- **Application**
- **Middleware, libraries**
- **Operating system**
- **BootROM, drivers, HAL**

- **Gravity, atmospheric, and propagator models**
- **Integrates with models from other vendors**

- **Controlled Environment**
- **Eclipse-based**

- **Human user interface**

Copyright © 2007 Virtutech, All rights reserved.
Network Interface Components

- Allows simulation to be distributable
- DDS - Generic Reconfigurable Interface Manager (GRIM) NDDS Interface
  - Allows distribution using several difference services
REATSS Capabilities

• Eclipse perspectives

• SDE – Service Development Environment
  – Critical for creating REATSS capable simulation
    ▪ Define simulation components (services)
      o Interfaces
      o Requirements
      o Visual Studio build files
    ▪ Eclipse based

• ADE – Application Development Environment
  – Allows you to plug services/components together
    ▪ Drag ‘n drop
    ▪ Displays various component metadata

• SCM – Simulation Control and Modeling
  – Start/pause/stop simulation
  – Breakpoints, checkpoints, and fault injection
  – Python-base scripting engine (automate all tests)
Flight Software Testing

- **Avionics Test Bench**
  - Uses real hardware
  - Most accurate, most expensive

- **Functional Test Bed**
  - Hybrid between hardware/simulation
  - Still expensive. Limited control/visibility

- **Engineering Simulation**
  - Completely software simulation based
  - Lower-fidelity than real hardware
  - Complete control/visibility
  - Lower cost and extremely scalable
Conclusions

- Integrated NDDS communications backplane
- Rapidly reconfigurable architecture
- Systems management dashboard

- All technologies extensible to other target systems

- REATSS enables you to
  “Test as you fly, fly as you test”
Contact Us

Chris Hall

chris@virtutech.com

949-683-4255