

Taller Introductorio a Simics

Instalación y Performance

I. ANTES DE EMPEZAR

Antes de iniciar se recomiendan los siguientes puntos:

- Leer la documentación de Simics que puede encontrar en el siguiente enlace de [documentación](#)
- Si desea motivación extra, vea el siguiente vídeo de Simics en la industria: [Nasa](#)

I-A. Requisitos

- Aunque Simics es multi plataforma, para efectos de este curso, se espera que el estudiante utilice Linux y se recomienda Ubuntu. El uso de otros sistemas operativos (Windows) no será soportado y se convierte en responsabilidad del estudiante.
- Se recomienda contar con un rango de memoria libre entre 50 GB y 100 GB para asegurar una instalación completa y exitosa.

I-B. Opciones para Linux

Si no posee una computadora con Linux, se tienen varias opciones:

- Use un [LiveUSB](#) para no modificar su computadora.
- Instale un software de virtualización para virtualizar el Linux donde hará el trabajo.
- Instale Linux en su máquina de trabajo y maneje varios sistemas operativos de ser necesario.

II. INSTALACIÓN

Luego, en su máquina de Linux, use el siguiente enlace y siga las instrucciones: [Cómo instalar Simics](#). El link de la página de descarga se encuentra abajo en esa página. Favor leer esa información antes de continuar.

En el caso de escoger la opción de la máquina virtual, obviar el siguiente paso: “Enable the Use of Virtualization”. Puede intentarlo, pero podría no funcionar dado a que existirían dos niveles de virtualización (uno del Linux y otro el que Simics usará) por lo que el driver de VMP podría no funcionar. Para los casos donde no hay virtualización anidada, pueden seguirse esos casos para que Simics sea más rápido. No obstante, no es un requisito para instalar Simics.

Luego de seguir los pasos de instalación, seleccione la pestaña addons en ispm:

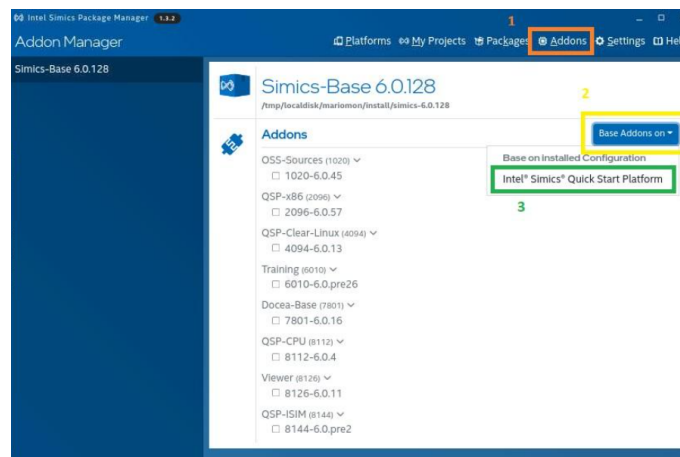


Figura 1: Simics Addons ipsm

Presione la opción de “Base Addons on” y seleccione “Intel Simics Quick Start Platform” (Ver figura 1). Si omite este paso, los paquetes extra, además del simulador base, no serán automáticamente traídos a su “espacio de trabajo”. Deberían de ver una pantalla como la siguiente (con todos los addons y una cajita con un check azul como lo indica en la imagen) y presione “Save updates”:

Nota: Tenga en cuenta la ruta señalada en amarillo a como se muestra en la Figura 2, porque necesitara de esa ruta más adelante en el proyecto (importante para la sección del ejercicio)

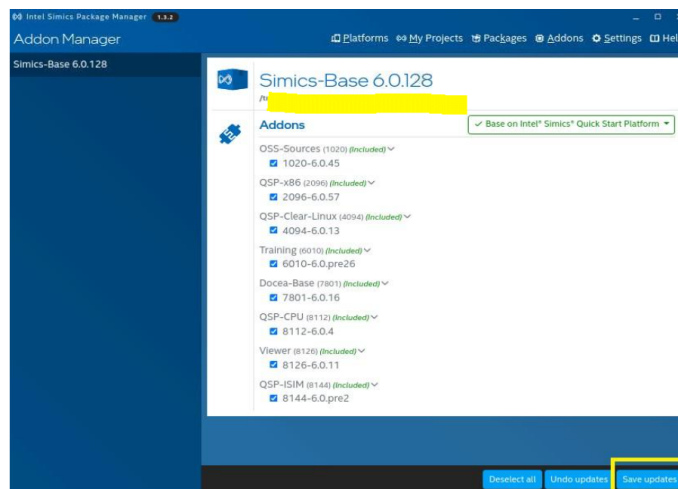


Figura 2: Intel Simics Quick Start Platform

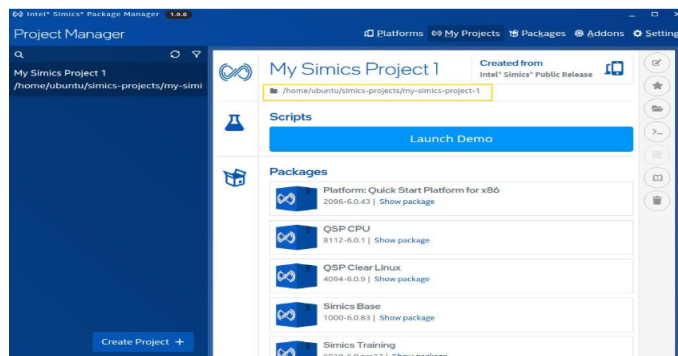


Figura 3: Enter Caption

En la imagen superior, se señaló con una caja amarilla la ruta donde estará su proyecto de Simics (use “cd” en la terminal o investigue cómo usar cd si no lo sabe). Ejecute simics desde la carpeta de su espacio de trabajo: `./simics` (esto iniciará una pantalla de control y podrá interactuar con lo referido en el documento como CLI).

III. FAMILIARIZARSE CON EL USO DE SIMICS

Procederemos a usar algunos comandos de Simics, guiados por el entrenamiento incluido en el producto.

1. Busque en su proyecto, la carpeta `doc/`. Aquí está el entrenamiento oficial de Simics. Siéntase en la libertad de revisarlo. Realizaremos el laboratorio de `nut002-Simics-CLI.pdf`.
2. Recuerde, no se usará ECLIPSE en la tarea. Por lo tanto, ciertas partes del laboratorio no le serán útiles por lo cual las ignoraremos. A continuación, se nombrarán las partes a ignorar e indicaciones adicionales:
 - a. Página 3. Para iniciar una sesión de Simics utilice la línea de comandos en una terminal (Ver Figura 3):
 - b. Página 11. Ignore “Simics command-line output text encoding and log files in editors”. Retome el laboratorio en la página 13
 - c. Página 17. Ignore las referencias en el paso 10 al IDE de eclipse. Revise los archivos con “ls” en la carpeta de Simics, por ejemplo.
 - d. Página 20. STOP! Listo, si desea continuar puede hacerlo, pero no es necesario.

IV. EJERCICIO

Simics puede ser usado para hacer prototipado de dispositivos y simulación de una plataforma (tarjeta madre, periféricos, SoCs). Esto posteriormente es usado para la validación y desarrollo de software para toda la pila de Software: firmwares, BIOS/UEFI, el sistema operativo y aplicaciones de usuario. Por esto, incluye maneras de explorar a fondo cada una de las instrucciones, escrituras en memoria y hacer “profiling” de aplicaciones. Haremos unos ejercicios para hacer profiling de dos algoritmos programados en C++. Usaremos la instrucción mágica para determinar el momento exacto en que empieza la ejecución de los métodos y usaremos el “instrumentation framework” de Simics para analizar la ejecución de ambos algoritmos.

1. Entre a la siguiente página <https://github.com/mmongeo/academia-collaboration>
2. El README tiene pasos para descargar una “craff image”. Los archivos .craff son archivos comprimidos de imágenes de discos duros. Esta será la imagen de disco que será la usada en el ambiente simulado.
3. Descargue el repositorio del primer paso.
4. Muévase a la carpeta que dice simics-project del repositorio y ejecute el binario desde la carpeta de instalación de Simics de esta manera al ejecutable project-setup:
 - a. `cd simics-project`
 - b. `../instalation-directory/simics-6.X.X/bin/project-setup --ignore-existing-files`
 - c. Esto crea un nuevo espacio de trabajo y la habilidad de usar algunos archivos del repositorio para el ejercicio.
5. Asegúrese que puede ejecutar simics correctamente desde la carpeta simics-project, ejecutando `./simics`
 - a. Si no tiene problemas, solo salga de la simulación.
 - b. De lo contrario revise el mensaje del error o contacte a su profesor o a alguno de los miembros de Intel
6. Compile la documentación ejecutando el ejecutable en la carpeta simics-project:
 - a. `./documentation`
 - b. Eso automaticamente abre la documentacion en el navegador
 - c. Lea sobre el instrumentation framework

Simics User's Guide / II Feature Overview /

9 Instrumentation

- c. Lea sobre el magic instruction

12.1.7 Magic Instructions and Magic Breakpoints

7. Explore la carpeta sort-magic. Lea y entienda cómo opera el programa en c++ y los demás archivos incluidos.
8. Modifique la primera línea no comentada del archivo `target/qsp-x86/sort-magic.simics` y apunte el parámetro `disk0-image` con la ruta del craff que descargó en el paso 2.
9. Ejecute `./simics targets/qsp-x86/sort-magic.simics`
10. Observe los mensajes en la consola de Simics, este lo guiará en el ejercicio. Si desea repetirlo, solo cierre la simulación y vuelva a hacer el paso 9.

V. CUESTIONARIO

1. ¿Qué hace el código del `sort-magic.cpp` en las líneas `MAGIC(..)`, según lo leído en el paso 6?
2. ¿Cómo esto se relaciona con el archivo `targets/qsp-x86/sort-magic.simics`? Utilice el comando `help` desde la consola de Simics donde lo necesite, una pista es: `help bp.magic.wait-for`
3. ¿Cuál es la complejidad de los algoritmos Quicksort y BubbleSort en notación O? ¿En promedio, cuál de los dos algoritmos debería de ejecutarse más rápidamente?
4. Guarde los histogramas de ambos algoritmos y adjúntelos en el entregable.
5. ¿Sabiendo la complejidad de ambos algoritmos, cuál es el impacto en la cantidad de instrucciones ejecutadas en el rendimiento? Es decir, elabore en cuáles operaciones del procesador deberían de ser más costosas y por ende hacer que el programa tome más tiempo y donde hay la mayor diferencia entre ambos histogramas (un gráfico le ayudará a comparar mejor los valores).
6. ¿Cuál de las dos implementaciones de ordenamiento elegiría para un programa y por qué?

VI. CONTACTOS

Ante cualquier consulta, no duden en contactarnos.

- Lucía Monge: m.lucia.monge.golcher@intel.com
- Sebastián Yock: sebastian.calderon.yock@intel.com — @CvazCY (Telegram)

VII. TABLA DE EVALUACIÓN

Secciones	Escala de calificación			
Categorías	Deficiente 0 Puntos	Regular 1 Punto	Bien 2 Puntos	Excelente (3Puntos)
Ejercicios prácticos (Peso 30)	No logra setear el entorno de trabajo ni ejecutar Simics.	Logra setear el entorno de trabajo, pero no ejecutar Simics.	Logra setear el entorno de trabajo y ejecutar Simics. No compila ni ejecuta los programas.	Logra setear el entorno de trabajo, ejecutar Simics; así como compilar y ejecutar los programas.
Uso de la herramienta Simics. Debe presentar las capturas (Peso 30)	No ejecuta la simulación con una imagen .craff	Utiliza una imagen .craff para ejecutar la simulación. No utiliza comandos de Instrumentación, Magic Instructions ni generación de histogramas.	Utiliza una imagen .craff para ejecutar la simulación. Utiliza algunos de los comandos de Instrumentación y Magic Instructions, pero no genera histogramas.	Utiliza una imagen .craff para ejecutar la simulación. Utiliza los comandos de Instrumentación, Magic Instructions y generación de histogramas.
Análisis (Peso 40)	No se realiza un análisis de lo solicitado.	El análisis mostrado está incompleto.	El análisis mostrado carece de profundidad. Sólo se menciona, pero no se explica.	Realiza un análisis detallado de lo solicitado.