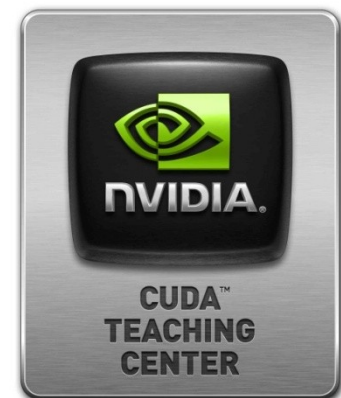


# Computación de Propósito General en Unidades de Procesamiento Gráfico (GPGPU)

E. Dufrechou, P. Ezzatti y M. Pedemonte



# **Clase 10**

## **Ecosistema CUDA (extendido)**

# Debugging y Profiling

# Debugging

Distintas posibilidades para debugging de CUDA:

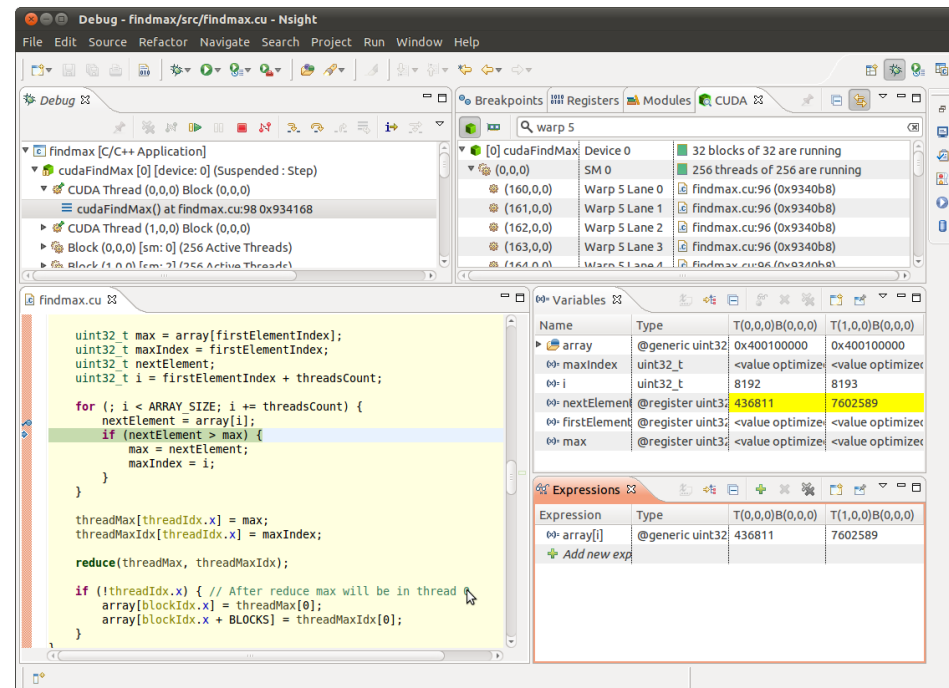
- CUDA-GDB debugger de línea de comandos (Linux y Mac)
  - es una extensión de GDB x86-64 (the GNU Project debugger).
  - NVIDIA Parallel NSight incorpora las funcionalidades en la IDE
  - Wrappers gráficos: Data Display Debugger (DDD), EMACS

- Otros debuggers para CUDA:

- ARM DDT (  
<https://developer.nvidia.com/allinea-ddt>  
)
- TotalView (<https://totalview.io/>)

- CUDA-MEMCHECK

- Útil para detectar errores en el acceso a memoria



# CUDA-GDB

**Funcionamiento similar a GDB:**

- **Debugging simultáneo en CPU y GPU**
- **Breakpoints**
- **Inspección de memoria, registros, variables en memoria local/shared/global**
- **Soporta múltiples GPUs, contextos, y kernels**
- **Debugging a nivel de código fuente y ensamblador (SASS)**

# CUDA-GDB

**Funcionamiento similar a GDB:**

- **Debugging simultáneo en CPU y GPU**
- **Breakpoints**
- **Inspección de memoria, registros, variables en memoria local/shared/global**
- **Soporta múltiples GPUs, contextos, y kernels**
- **Debugging a nivel de código fuente y ensamblador (SASS)**

# CUDA-GDB

Algunos comandos:

- run, continue, kill
- next (no entra en funciones), step (entra en funciones)
- break (pone un breakpoint)
  - break my\_kernel, break \_Z6kernelIfiEvPT\_PT0, break acos.cu:380, break \*0x3e840a8, break \*\$pc, set cuda break\_on\_launch application
- cuda (cambia el foco a un kernel/bloque/thread específico)
  - cuda kernel 2 block 1,0,0 thread 3,0,0
- info cuda devices/kernels/threads (muestra devices, kernels, threads activos)
  - info cuda threads kernel 2
- print (leer o escribir variables/direcciones de memoria)
  - print var, print &var, print var = 3

# CUDA-MEMCHECK

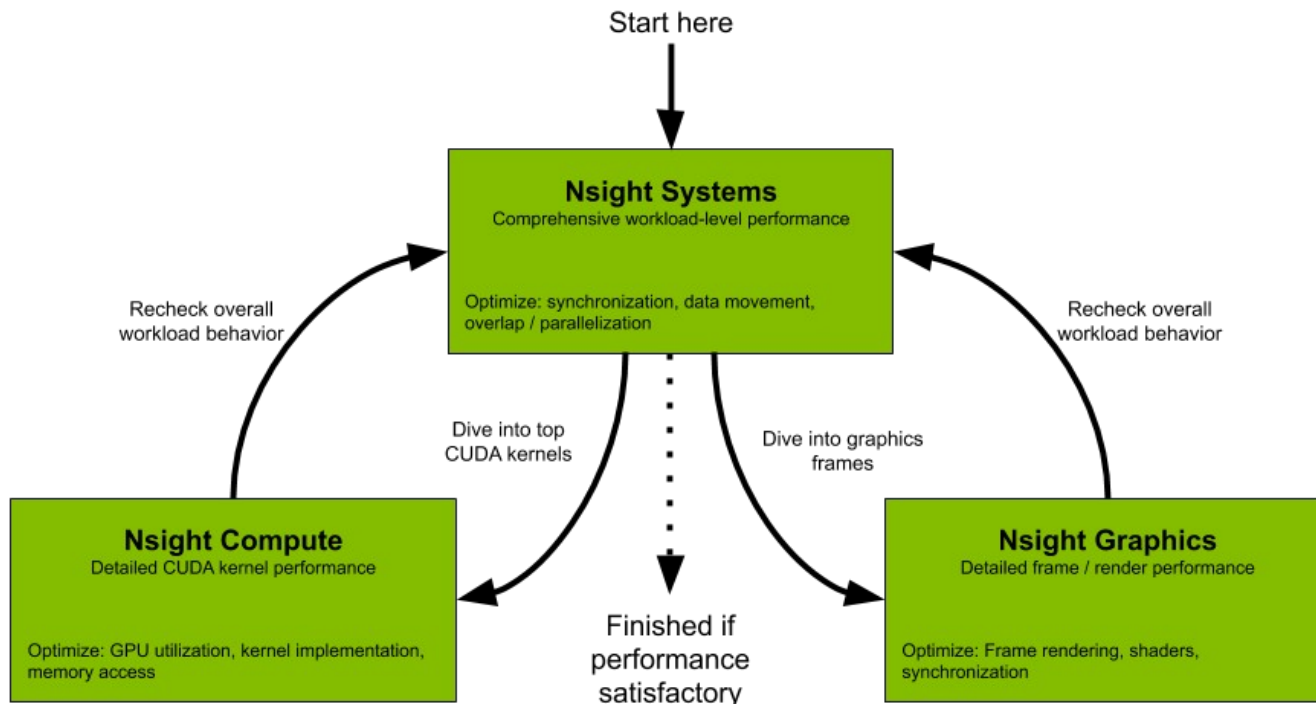
**Dos formatos:**

- **Standalone**
  - `cuda-memcheck [op] programa params`
- **Integrado con cuda-gdb (Linux y Mac)**
  - `set cuda memcheck on`
- **Util para chequear condiciones de carrera:**
  - `cuda-memcheck --tool racecheck programa params`



# Profiling

- En el curso ya usamos nvprof (actualmente deprecada)
- Nuevas herraminetas: NSight Systems y NSight Compute
  - [https://developer.nvidia.com/gameworksdownload#?tx=\\$gameworks,developer\\_tools](https://developer.nvidia.com/gameworksdownload#?tx=$gameworks,developer_tools)
  - Ambas cuentan con GUIs y clientes de línea de comandos



# NSight Systems

- Nos da una visión de todo el sistema:
  - CPUs, GPUs, hilos de CPU, streams en GPU
  - Ejecución de kernels, transferencias de memoria, etc.
- Nos muestra los kernels ejecutados, con los tamaños de grilla, recursos utilizados, etc.
- Puede generarse un archivo en línea de comandos y luego importarlo en la GUI



# NSight Systems

- Herramienta de línea de comandos: `nsys`
- Dos modos:
  - automático (`profile`)
  - interactivo (`start,stop,launch,etc...`)
- Ejemplo automático:  
`nsys profile -o salida -f true ./prog params`
  - Genera archivo `salida.nsys-rep` importable en NSight Systems
- Ejemplo interactivo:  
`nsys start --stop-on-exit=false`  
`nsys launch --trace=cuda --sample=none ./prog params`  
`nsys stop`
- `--stats` muestra un resumen de estadísticas en formato de texto plano

# NSight Compute

- Herramienta de línea de comandos: ncu
- Tres modos:
  - Launch-and-attach (profile)
  - Launch / attach (attach permite hacer profiling remoto)
- Ejemplo launch-and-attach (igual que nsys):
  - `nsys profile -o salida -f true ./prog params`
  - Genera archivo importable en NSight Compute
  - Por defecto extrae un conjunto pequeño de métricas
  - `ncu --list-sets` (lista los sets de métricas disponibles)
  - `ncu -o prof_ncu -f --set full`

# Recursos

- **Descargar herramientas (es necesario crear una cuenta en Nvidia Dev Zone):**
  - <https://developer.nvidia.com/gameworksdownload>
- **Documentación:**
  - <https://docs.nvidia.com/cuda/cuda-gdb/>
  - <https://docs.nvidia.com/nsight-systems/>
  - <https://docs.nvidia.com/nsight-compute/>
- **Videos:**
  - <https://www.youtube.com/user/NVIDIADeveloper>