PROFILING

February 28, 2020

NSIGHT DEVELOPER TOOLS

NSIGHT PRODUCT FAMILY

Standalone Performance Tools

Nsight Systems - System-wide application algorithm tuning
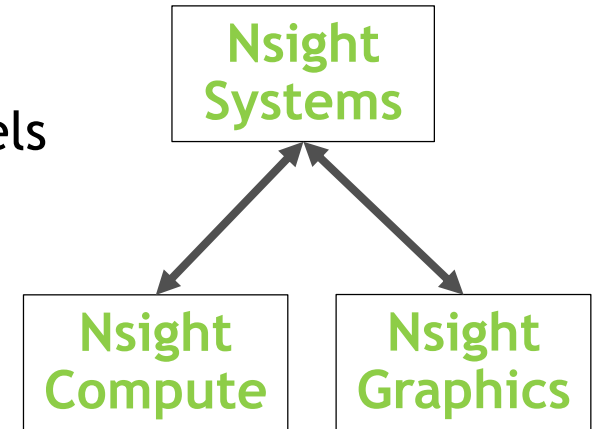
Nsight Compute – Debug CUDA API and optimize CUDA kernels

Nsight Graphics - Debug/optimize specific graphics apps

IDE Plugins

Nsight Eclipse Edition/Visual Studio – editor, debugger, some perf analysis

Workflow

```
        Nsight
        Systems
         ↗   ↘
   Nsight      Nsight
   Compute     Graphics
```

NVIDIA.

NSIGHT SYSTEMS
System Profiling Tool

System-wide application algorithm tuning
    Multi-process tree support

Locate optimization opportunities
    Visualize millions of events on a fast GUI timeline
    Or gaps of unused CPU and GPU time

Balance your workload across multiple CPUs and GPUs
    CPU algorithms, utilization, and thread state
    GPU streams, kernels, memory transfers, etc

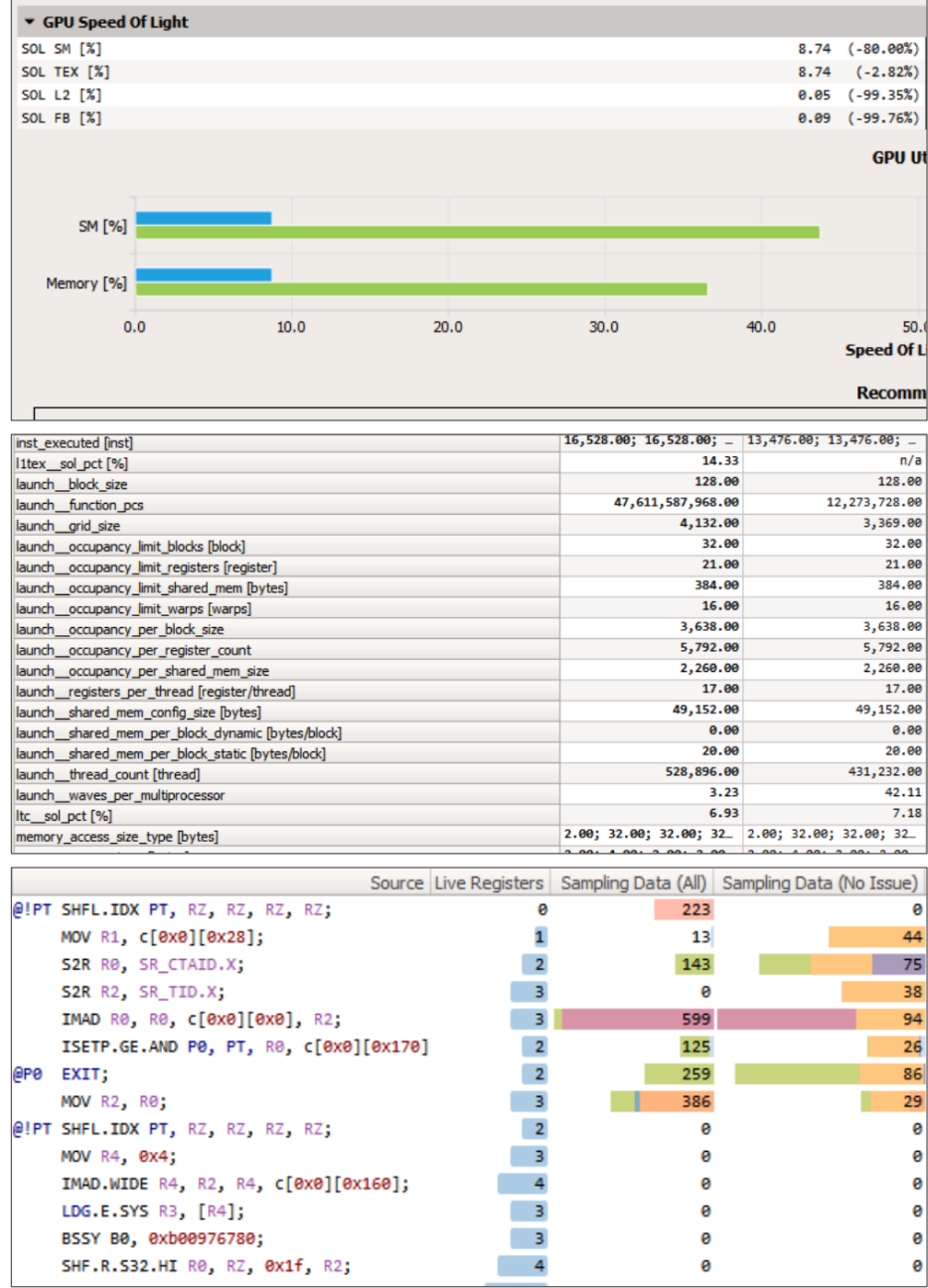Multi-platform: Linux, Windows, Mac OS X (host only)
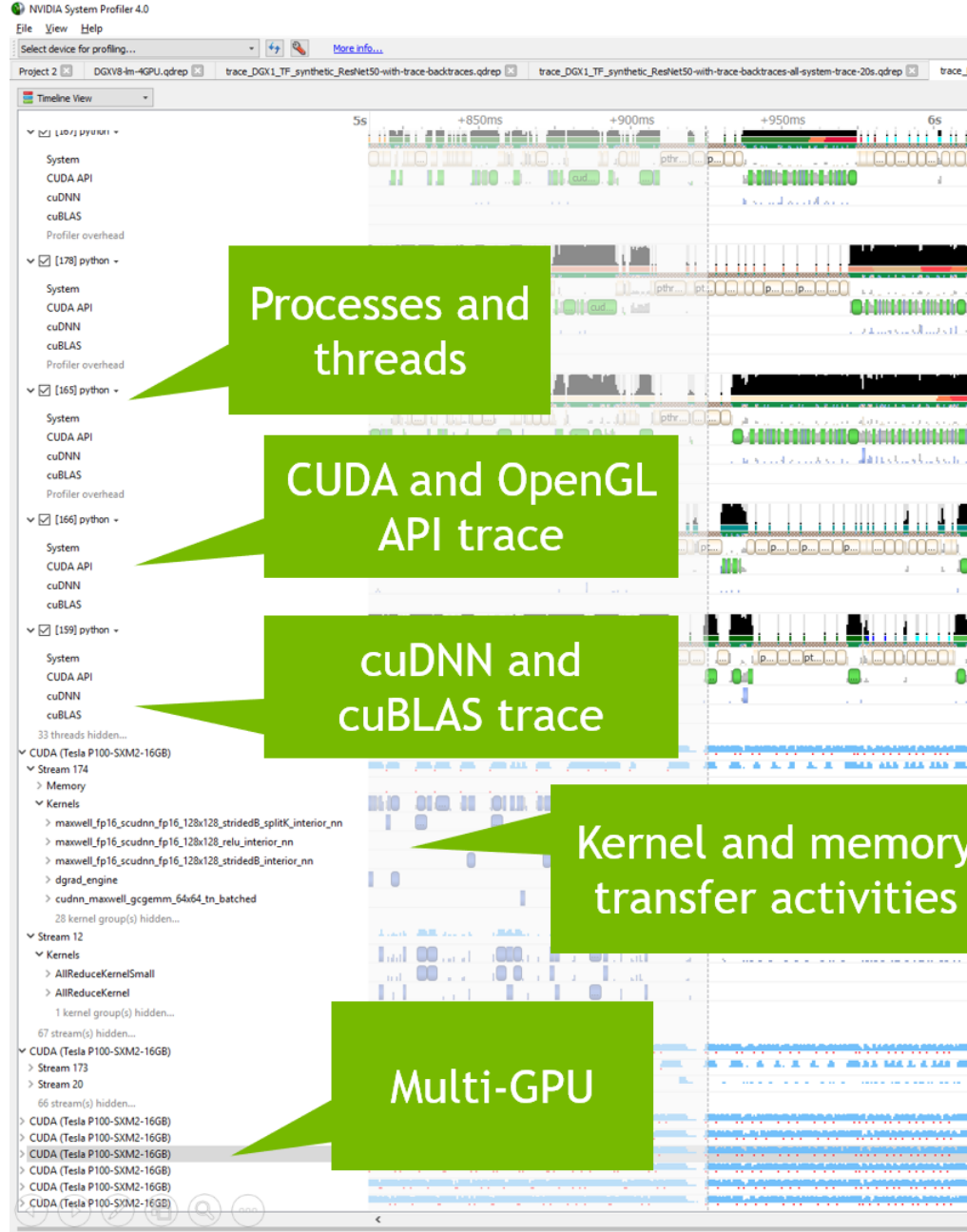
NSIGHT COMPUTE

Next-Gen Kernel Profiling Tool

Key Features:

- Interactive CUDA API debugging and kernel profiling

- Fast Data Collection

- Improved Workflow (diffing results)

- Fully Customizable (programmable UI/Rules)

- Command Line, Standalone, IDE Integration

OS: Linux, Windows, Mac OS X (host only)

GPUs: Volta, Turing

▼ GPU Speed Of Light

SOL SM [%]	8.74	(-80.00%)
SOL TEX [%]	8.74	(-2.82%)
SOL L2 [%]	0.05	(-99.35%)
SOL FB [%]	0.09	(-99.76%)

GPU Ut

SM [%]

Memory [%]

0.0   10.0   20.0   30.0   40.0   50.0

Speed Of L

Recomm

inst_executed [inst]	16,528.00; 16,528.00; ...	13,476.00; 13,476.00; ...
l1tex__sol_pct [%]	14.33	n/a
launch__block_size	128.00	128.00
launch__function_pcs	47,611,587,968.00	12,273,728.00
launch__grid_size	4,132.00	3,369.00
launch__occupancy_limit_blocks [block]	32.00	32.00
launch__occupancy_limit_registers [register]	21.00	21.00
launch__occupancy_limit_shared_mem [bytes]	384.00	384.00
launch__occupancy_limit_warps [warps]	16.00	16.00
launch__occupancy_per_block_size	3,638.00	3,638.00
launch__occupancy_per_register_count	5,792.00	5,792.00
launch__occupancy_per_shared_mem_size	2,260.00	2,260.00
launch__registers_per_thread [register/thread]	17.00	17.00
launch__shared_mem_config_size [bytes]	49,152.00	49,152.00
launch__shared_mem_per_block_dynamic [bytes/block]	0.00	0.00
launch__shared_mem_per_block_static [bytes/block]	20.00	20.00
launch__thread_count [thread]	528,896.00	431,232.00
launch__waves_per_multiprocessor	3.23	42.11
ltc__sol_pct [%]	6.93	7.18
memory_access_size_type [bytes]	2.00; 32.00; 32.00; 32...	2.00; 32.00; 32.00; 32...

| Source	Live Registers	Sampling Data (All)	Sampling Data (No Issue)
@!PT SHFL.IDX PT, RZ, RZ, RZ, RZ;	0	223	0
MOV R1, c[0x0][0x28];	1	13	44
S2R R0, SR_CTAID.X;	2	143	75
S2R R2, SR_TID.X;	3	0	38
IMAD R0, R0, c[0x0][0x0], R2;	3	599	94
ISETP.GE.AND P0, PT, R0, c[0x0][0x170]	2	125	26
@P0 EXIT;	2	259	86
MOV R2, R0;	3	386	29
@!PT SHFL.IDX PT, RZ, RZ, RZ, RZ;	2	0	0
MOV R4, 0x4;	3	0	0
IMAD.WIDE R4, R2, R4, c[0x0][0x160];	4	0	0
LDG.E.SYS R3, [R4];	3	0	0
BSSY B0, 0xb00976780;	3	0	0
SHF.R.S32.HI R0, RZ, 0x1f, R2;	4	0	0

USING NSIGHT SYSTEMS

COLLECT A PROFILE WITH NSIGHT SYSTEMS

```
$ nsys profile --stats=true ./myapp.exe
```

Generated file: `report.qdrep`

Import for viewing into the Nsight Systems UI

The Nsight Systems UI can also be used for interactive system profiling

NVIDIA.
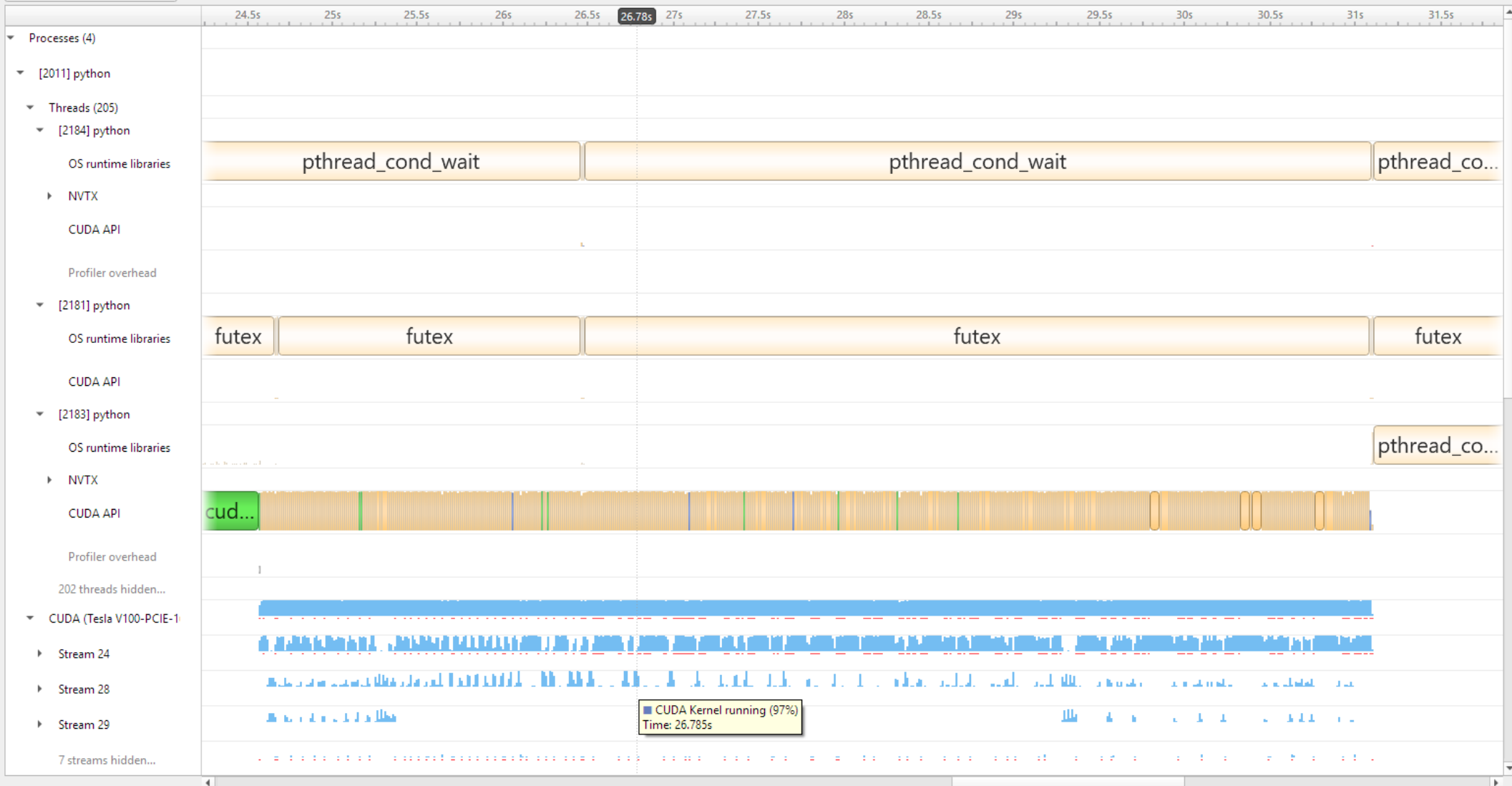
USING NSIGHT COMPUTE

KERNEL PROFILES WITH NSIGHT COMPUTE

```
$ nv-nsight-cu-cli –k mykernel ./myapp.exe
```

(Without the –k option, Nsight Compute will profile everything and take a long time!)

The Nsight Compute UI can also be used for interactive kernel profiling

NSIGHT COMPUTE UI

Current | 47828 - volta_fp16_s884cudnn_fp16_256x128_ldg8_relu_filter1x1_stg8_interior_nchw_nn_v1 (802816, 2, 1) **Time:** n/a **Cycles:** n/a **Regs:** 226 **GPU:** Tesla V100-PCIE-16GB **SM Frequency:** n/a **CC:** 7.0

▼ GPU Speed Of Light

SOL SM [%]	27.64	Duration [msecond]	1.20
SOL Memory [%]	56.82	Elapsed Cycles [cycle]	1,504,410
SOL TEX [%]	53.74	SM Active Cycles [cycle]	1,459,787.71
SOL L2 [%]	22.78	SM Frequency [cycle]	1,248,295,701.36
SOL FB [%]	42.09	Memory Frequency [cycle]	881,625,488.68

GPU Utilization

Recommendations

Bottleneck High-level bottleneck detection | Apply

▶ Compute Workload Analysis

Executed Ipc Elapsed [cycle]	1.11	SM Busy [%]	28.44
Executed Ipc Active [cycle]	1.14	Issue Slots Busy [%]	28.44
Issued Ipc Active [cycle]	1.14	Issue Slots Max [%]	29.02

▶ Memory Workload Analysis

Memory Throughput [Gbyte]	477.73	Mem Busy [%]	52.23
L1 Hit Rate [%]	13.35	Max Bandwidth [%]	56.82
L2 Hit Rate [%]	69.23	Mem Pipes Busy [%]	15.33

▼ Scheduler Statistics

Active Warps Per Scheduler [warp]	2.01	Instructions Per Active Issue Slot [inst/issue]	1
Eligible Warps Per Scheduler [warp]	0.40	No Eligible [%]	71.48
Issued Warp Per Scheduler [cycle]	0.29	One or More Eligible [%]	28.52