Debugging and Profiling in Microsoft VisualStudio with Parallel Nsight

GPU TECHNOLOGY CONFERENCE

NVIDIA Parallel Nsight™

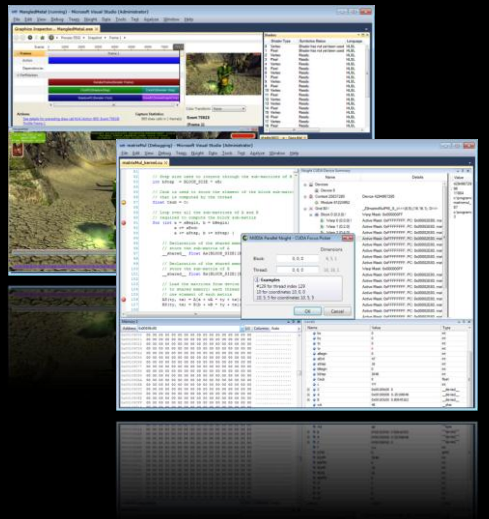
Visual Studio integrated development for GPU and CPU

Build **Debug** **Profile**

Microsoft® Visual Studio® PARTNER

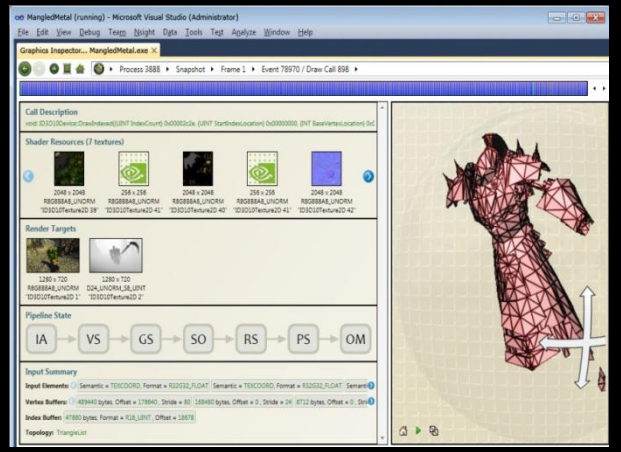
GPU TECHNOLOGY CONFERENCE

NVIDIA Parallel Nsight for Graphics & Compute



CUDA/Graphics Debugger

- ✓ GPU Accelerated CUDA and HLSL debugging
- ✓ Examine shaders executing in parallel
- ✓ Identify issues with conditional breakpoints

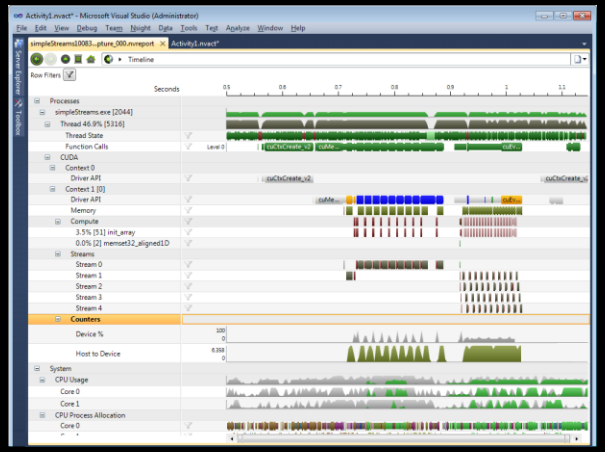


Graphics Inspector

- ✓ Real-time inspection of Direct3D API calls

Investigate GPU pipeline state

- ✓ See contributing fragments with Pixel History
- ✓ Profile frames to find GPU bottlenecks



System Analysis

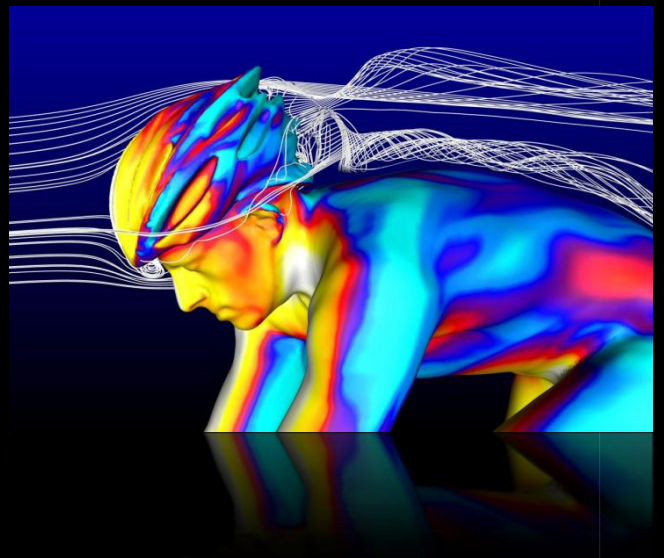
- ✓ View CPU & GPU events on a single timeline

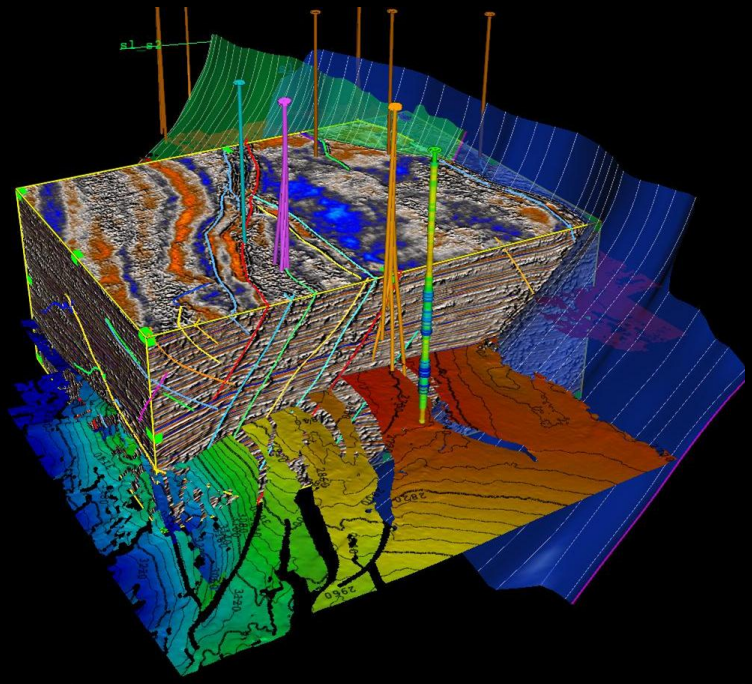
Examine workload dependencies

- ✓ CUDA, Direct3D, and OpenGL API Trace
- ✓ Profile CUDA kernels using performance counters

Free License!

Parallel Nsight for Compute Developers



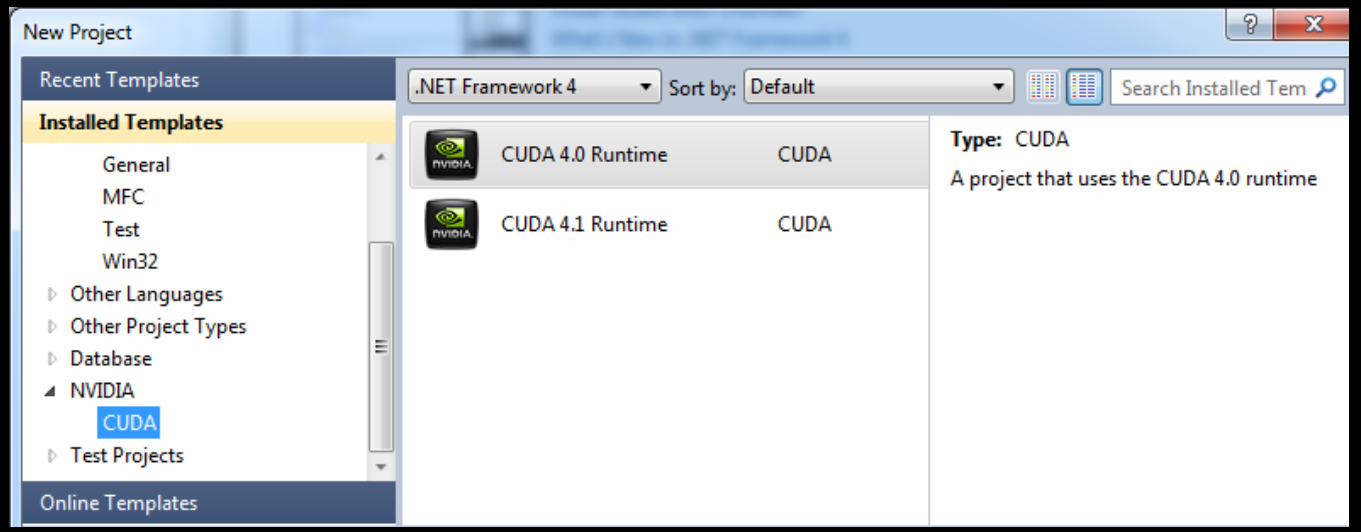


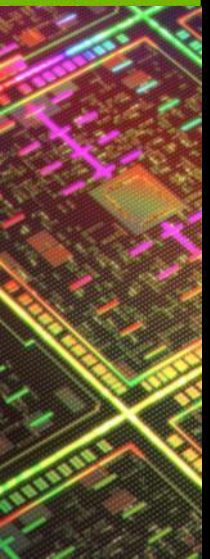
"With the advent of Parallel Nsight and CUDA support for debugging, the [CUDA] development process now more closely resembles that of traditional parallel CPU code"

Jacques du Toit – Scientific Computing

CUDA Build System

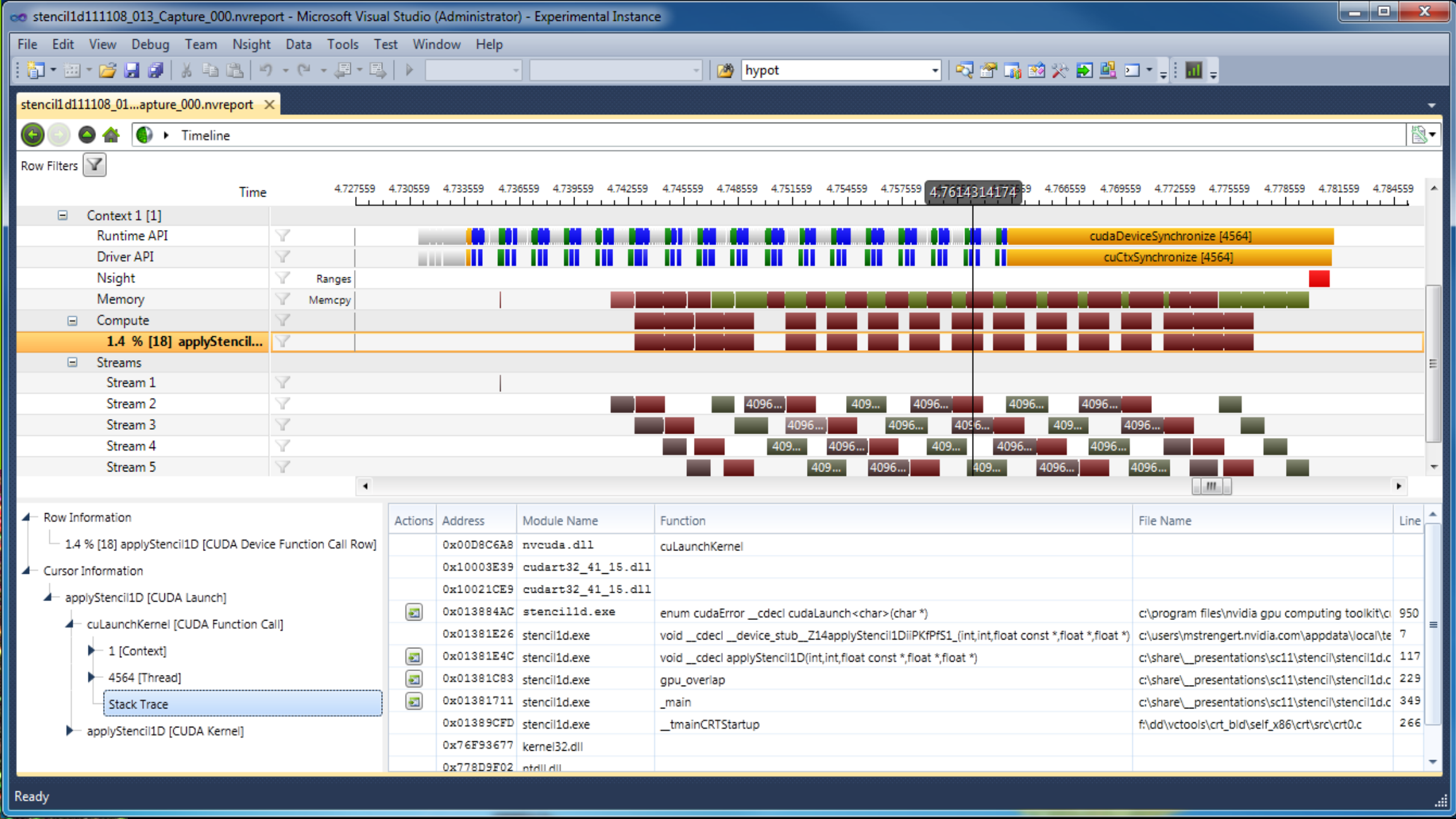
- Visual Studio 2008 SP1 and 2010 SP1
- CUDA project wizard
- Project setting extensions
- Visual C++ and .NET project integration

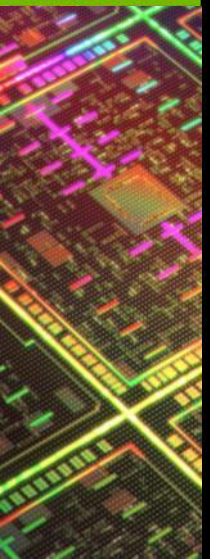




Parallel Nsight CUDA Debugging

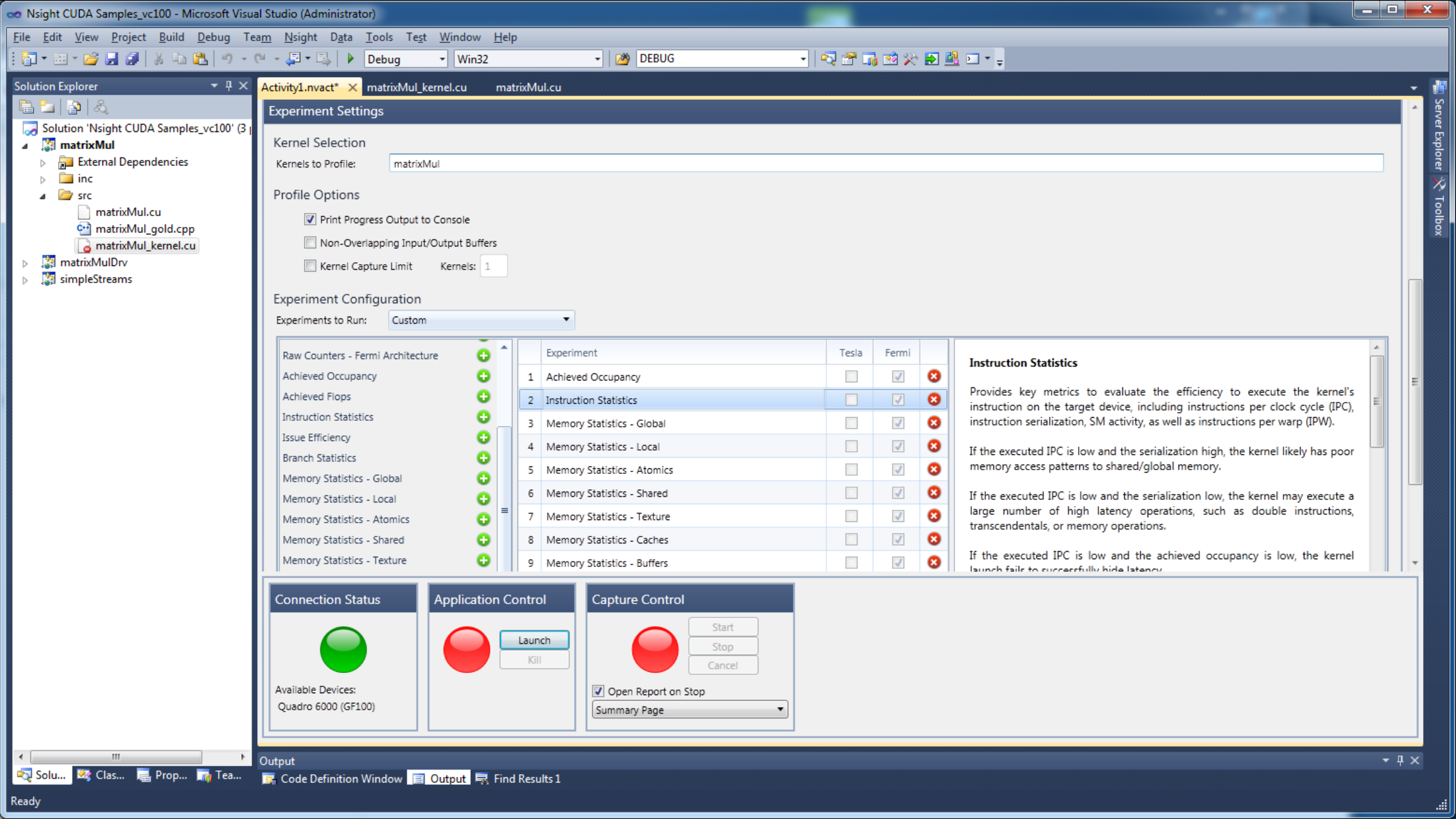
- Native GPU debugging with mixed CUDA-C/PTX/SASS assembly
- Debugger attach to running process
- Conditional breakpoint with program variables
- GPU memory views and data breakpoints
- CUDA expression engine and stack frame support
- Massively-threaded GPU kernels navigation
- CUDA memory checker
- CUDA system information


stencil1d111108_013_Capture_000.nvreport - Microsoft Visual Studio (Administrator) - Experimental Instance
File Edit View Debug Team Nsight Data Tools Test Window Help
hypot
stencil1d111108_01...apture_000.nvreport
Timeline
Row Filters
Time
4.727559 4.730559 4.733559 4.736559 4.739559 4.742559 4.745559 4.748559 4.751559 4.754559 4.757559 4.7614314174 4.766559 4.769559 4.772559 4.775559 4.778559 4.781559 4.784559
Context 1 [1]
Runtime API
Driver API
Nsight
Memory
Compute
1.4 % [18] applyStencil...
Streams
Stream 1
Stream 2
Stream 3
Stream 4
Stream 5
Ranges
Memcpy
cudaDeviceSynchronize [4564]
cuCtxSynchronize [4564]
4096...
409...
Row Information
1.4 % [18] applyStencil1D [CUDA Device Function Call Row]
Cursor Information
applyStencil1D [CUDA Launch]
cuLaunchKernel [CUDA Function Call]
1 [Context]
4564 [Thread]
Stack Trace
applyStencil1D [CUDA Kernel]
Actions Address Module Name Function File Name Line
0x00D8C6A8 nvcuda.dll cuLaunchKernel
0x10003E39 cudart32_41_15.dll
0x10021CE9 cudart32_41_15.dll
0x013884AC stencil1d.exe enum cudaError __cdecl cudaLaunch<char>(char *) c:\program files\nvidia gpu computing toolkit\c 950
0x01381E26 stencil1d.exe void __cdecl __device_stub__Z14applyStencil1DiiPKfPfS1_(int,int,float const *,float *,float *) c:\users\mstrengert.nvidia.com\appdata\local\te 7
0x01381E4C stencil1d.exe void __cdecl applyStencil1D(int,int,float const *,float *,float *) c:\share__presentations\sc11\stencil\stencil1d.c 117
0x01381C83 stencil1d.exe gpu_overlap c:\share__presentations\sc11\stencil\stencil1d.c 229
0x01381711 stencil1d.exe _main c:\share__presentations\sc11\stencil\stencil1d.c 349
0x01389CFD stencil1d.exe __tmainCRTStartup f:\dd\vctools\crt_bld\self_x86\crt\src\crt0.c 266
0x76F93677 kernel32.dll
0x778D9F02 ntdll.dll
Ready



Parallel Nsight CUDA Profiling

- CUDA profiler with live counter reconfiguration
- Unlimited experiments on live kernels
- Advanced profiling experiments
  - Achieved occupancy
  - Instruction throughput
- Kernel profiling filtering


Nsight CUDA Samples_vc100 - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Window Help
Debug
Win32
DEBUG
Solution Explorer
Solution 'Nsight CUDA Samples_vc100' (3
matrixMul
External Dependencies
inc
src
matrixMul.cu
matrixMul_gold.cpp
matrixMul_kernel.cu
matrixMulDrv
simpleStreams
Activity1.nvact*
matrixMul_kernel.cu
matrixMul.cu
Experiment Settings
Kernel Selection
Kernels to Profile:
matrixMul
Profile Options
Print Progress Output to Console
Non-Overlapping Input/Output Buffers
Kernel Capture Limit
Kernels:
1
Experiment Configuration
Experiments to Run:
Custom
Raw Counters - Fermi Architecture
Achieved Occupancy
Achieved Flops
Instruction Statistics
Issue Efficiency
Branch Statistics
Memory Statistics - Global
Memory Statistics - Local
Memory Statistics - Atomics
Memory Statistics - Shared
Memory Statistics - Texture
Experiment
Tesla
Fermi
1 Achieved Occupancy
2 Instruction Statistics
3 Memory Statistics - Global
4 Memory Statistics - Local
5 Memory Statistics - Atomics
6 Memory Statistics - Shared
7 Memory Statistics - Texture
8 Memory Statistics - Caches
9 Memory Statistics - Buffers
Instruction Statistics
Provides key metrics to evaluate the efficiency to execute the kernel's instruction on the target device, including instructions per clock cycle (IPC), instruction serialization, SM activity, as well as instructions per warp (IPW).
If the executed IPC is low and the serialization high, the kernel likely has poor memory access patterns to shared/global memory.
If the executed IPC is low and the serialization low, the kernel may execute a large number of high latency operations, such as double instructions, transcendentals, or memory operations.
If the executed IPC is low and the achieved occupancy is low, the kernel
Connection Status
Available Devices:
Quadro 6000 (GF100)
Application Control
Launch
Kill
Capture Control
Start
Stop
Cancel
Open Report on Stop
Summary Page
Output
Code Definition Window
Output
Find Results 1
Solu...
Clas...
Prop...
Tea...
Ready
Server Explorer
Toolbox

Nsight CUDA Samples_vc100 - Microsoft Visual Studio (Administrator)

File Edit View Project Build Debug Team Nsight Data Tools Test Window Help

Debug | Win32 | DEBUG

Solution Explorer

Solution 'Nsight CUDA Samples_vc100' (3
- matrixMul
  - External Dependencies
  - inc
  - src
    - matrixMul.cu
    - matrixMul_gold.cpp
    - matrixMul_kernel.cu
- matrixMulDrv
- simpleStreams

matrixMul111109_00...pture_000.nvreport | Activity1.nvact* | matrixMul_kernel.cu | matrixMul.cu

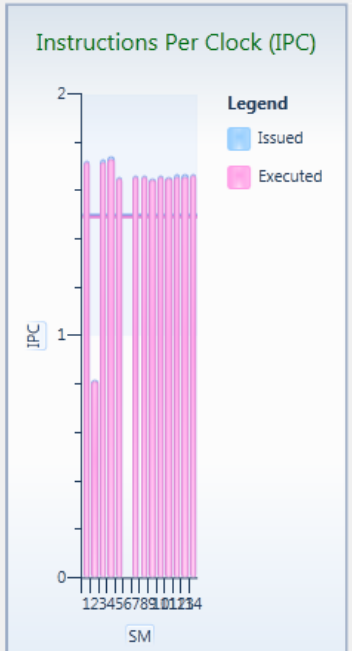
CUDA Launches

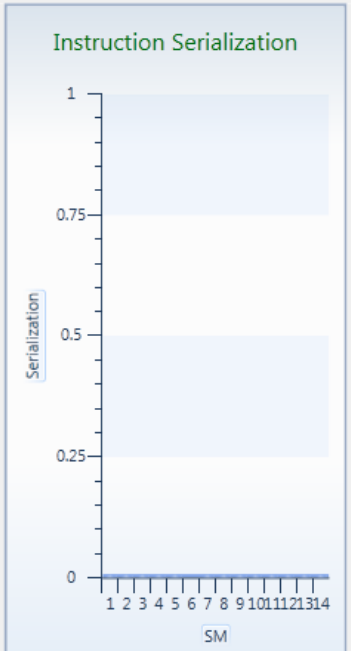
Filter — Viewing: 1 / 1

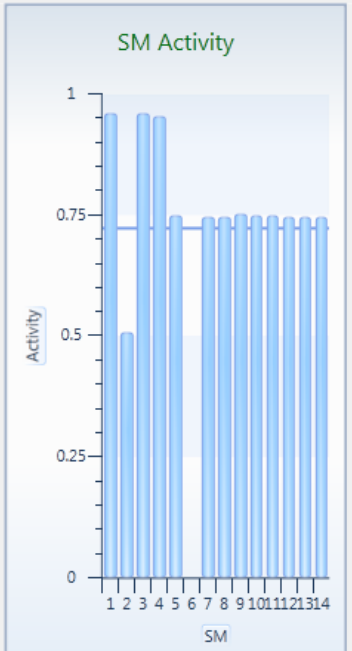
Drag a column header and drop it here to group by that column

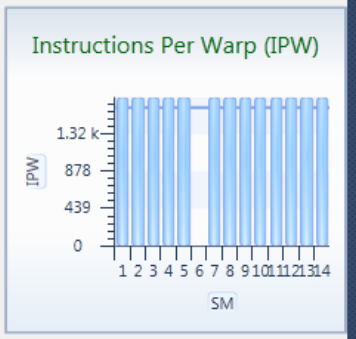
	Function Name	Module ID	Function ID	Start Time (µs)	Duration (µs)	Occupancy	Registers per Thread	Static Shared Memory per Block (bytes)	Dynamic Shared Memory per Block (bytes)	Local Memory per Thread (bytes)	Local Memory (bytes)	Cache Configuration Executed	Grid Dimensions	Block Dimensions
1	matrixMul	6	1	1,314,402.367	58.112	1.00	15	2048	0	0	43384832	PREFER_SHARED	{8, 5, 1}	{16, 16

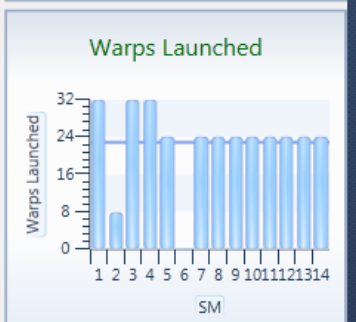
- matrixMul [CUDA Launch]
  - matrixMul [CUDA Kernel]
  - Experiment Results
    - CUDA Occupancy
    - CUDA Instruction Statistics
    - CUDA Memory Statistics











Output

Code Definition Window | Output | Find Results 1

Ready

Fully Featured Configurations…

- ✓ WDDM driver
- ✓ Tesla Compute Cluster driver

- ✓ Application and system trace
- ✓ CUDA profiling
- ✓ CUDA debugger
- ✓ CUDA memory checker

2 GPUs

Tesla + GPU

Remote PC

Multi-OS

Partially Featured Configuration…

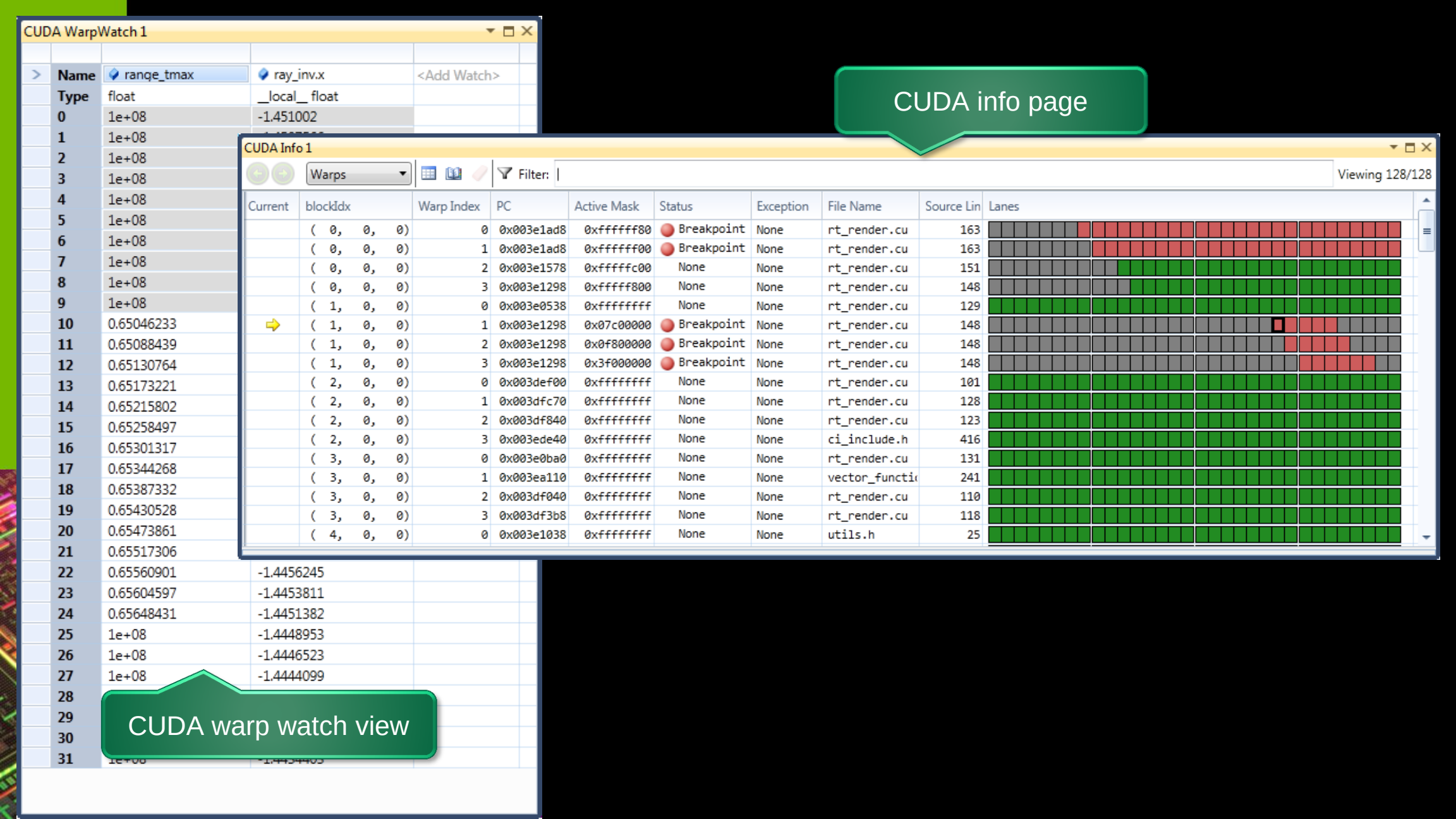
- ✓ WDDM driver
- ✓ Application and system trace
- ✓ CUDA profiling

Single GPU

New in Parallel Nsight 2.1

- CUDA Toolkit 4.1 support
- Support for CUDA C/C++ debugging on Optimus
- New CUDA and system information page
- CUDA parallel warp watch view
- GPU break on assert
- Debugging asynchronous kernel launches
- New CUDA profiling experiments
  - Flow control efficiency and branch divergence
  - Memory coalescing and cache efficiency
  - Statistics on issue dependencies and stall reasons
- OpenCL 1.1 API trace support

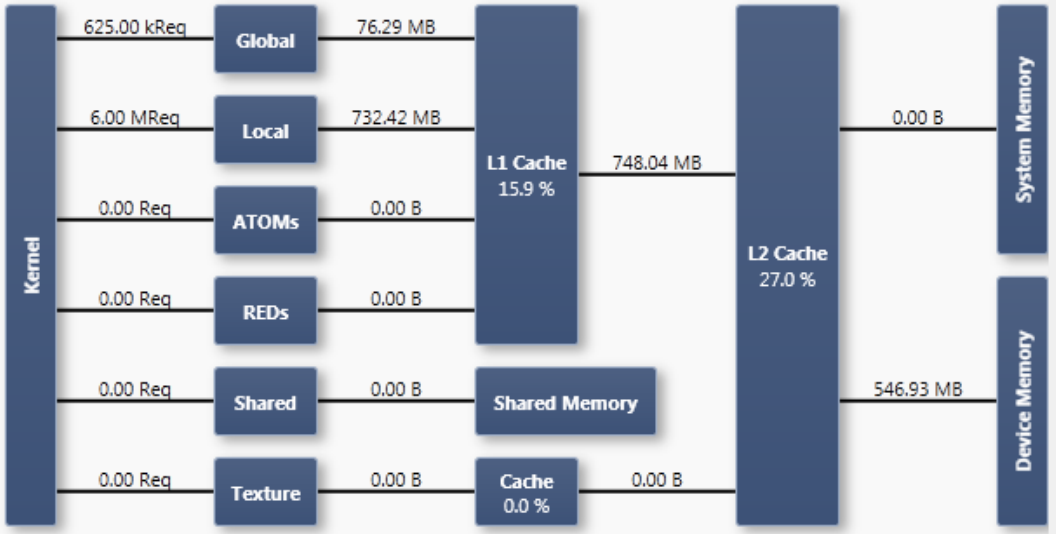
Available Q4 2011
RC2 available now!


CUDA WarpWatch 1
Name
range_tmax
ray_inv.x
<Add Watch>
Type
float
__local__ float
0 1e+08 -1.451002
1 1e+08
2 1e+08
3 1e+08
4 1e+08
5 1e+08
6 1e+08
7 1e+08
8 1e+08
9 1e+08
10 0.65046233
11 0.65088439
12 0.65130764
13 0.65173221
14 0.65215802
15 0.65258497
16 0.65301317
17 0.65344268
18 0.65387332
19 0.65430528
20 0.65473861
21 0.65517306
22 0.65560901 -1.4456245
23 0.65604597 -1.4453811
24 0.65648431 -1.4451382
25 1e+08 -1.4448953
26 1e+08 -1.4446523
27 1e+08 -1.4444099
28
29
30
31
CUDA warp watch view
CUDA info page
CUDA Info 1
Warps
Filter:
Viewing 128/128
Current
blockIdx
Warp Index
PC
Active Mask
Status
Exception
File Name
Source Lin
Lanes
(0, 0, 0) 0 0x003e1ad8 0xffffff80 Breakpoint None rt_render.cu 163
(0, 0, 0) 1 0x003e1ad8 0xffffff00 Breakpoint None rt_render.cu 163
(0, 0, 0) 2 0x003e1578 0xfffffc00 None None rt_render.cu 151
(0, 0, 0) 3 0x003e1298 0xfffff800 None None rt_render.cu 148
(1, 0, 0) 0 0x003e0538 0xffffffff None None rt_render.cu 129
(1, 0, 0) 1 0x003e1298 0x07c00000 Breakpoint None rt_render.cu 148
(1, 0, 0) 2 0x003e1298 0x0f800000 Breakpoint None rt_render.cu 148
(1, 0, 0) 3 0x003e1298 0x3f000000 Breakpoint None rt_render.cu 148
(2, 0, 0) 0 0x003def00 0xffffffff None None rt_render.cu 101
(2, 0, 0) 1 0x003dfc70 0xffffffff None None rt_render.cu 128
(2, 0, 0) 2 0x003df840 0xffffffff None None rt_render.cu 123
(2, 0, 0) 3 0x003ede40 0xffffffff None None ci_include.h 416
(3, 0, 0) 0 0x003e0ba0 0xffffffff None None rt_render.cu 131
(3, 0, 0) 1 0x003ea110 0xffffffff None None vector_functi 241
(3, 0, 0) 2 0x003df040 0xffffffff None None rt_render.cu 110
(3, 0, 0) 3 0x003df3b8 0xffffffff None None rt_render.cu 118
(4, 0, 0) 0 0x003e1038 0xffffffff None None utils.h 25

	Function Name	Module ID	Function ID	Start Time (µs)	Duration (µs)	Occupancy	Registers per Thread	Static Shared Memory per Block (bytes)	Dynamic Shared Memory per Block (bytes)	Local Memory per Thread (bytes)	Local Memory (bytes)	Cache Configuration Executed	Grid Dimensions	Block Dimensions	API Call ID
1	BlackScholesGPU	6	1	12,977,667.492	4,903.808	0.67	25	0	0	36	47579136	PREFER_SHARED	{480, 1, 1}	{128, 1, 1}	139
2	BlackScholesGPU	6	1	24,784,756.196	4,912.544	0.67	25	0	0	72	48365568	PREFER_SHARED	{480, 1, 1}	{128, 1, 1}	140

- BlackScholesGPU [CUDA Launch]
  - BlackScholesGPU [CUDA Kernel]
  - Experiment Results
    - CUDA Occupancy
    - CUDA Instruction Statistics
    - CUDA Branch Statistics
    - CUDA Issue Efficiency
    - CUDA Achieved Flops
    - CUDA Memory Statistics

View By: Size

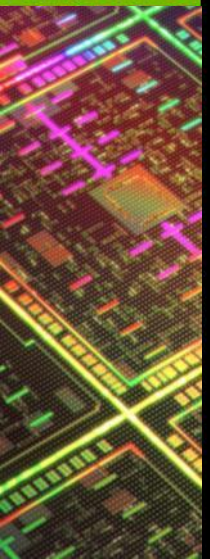


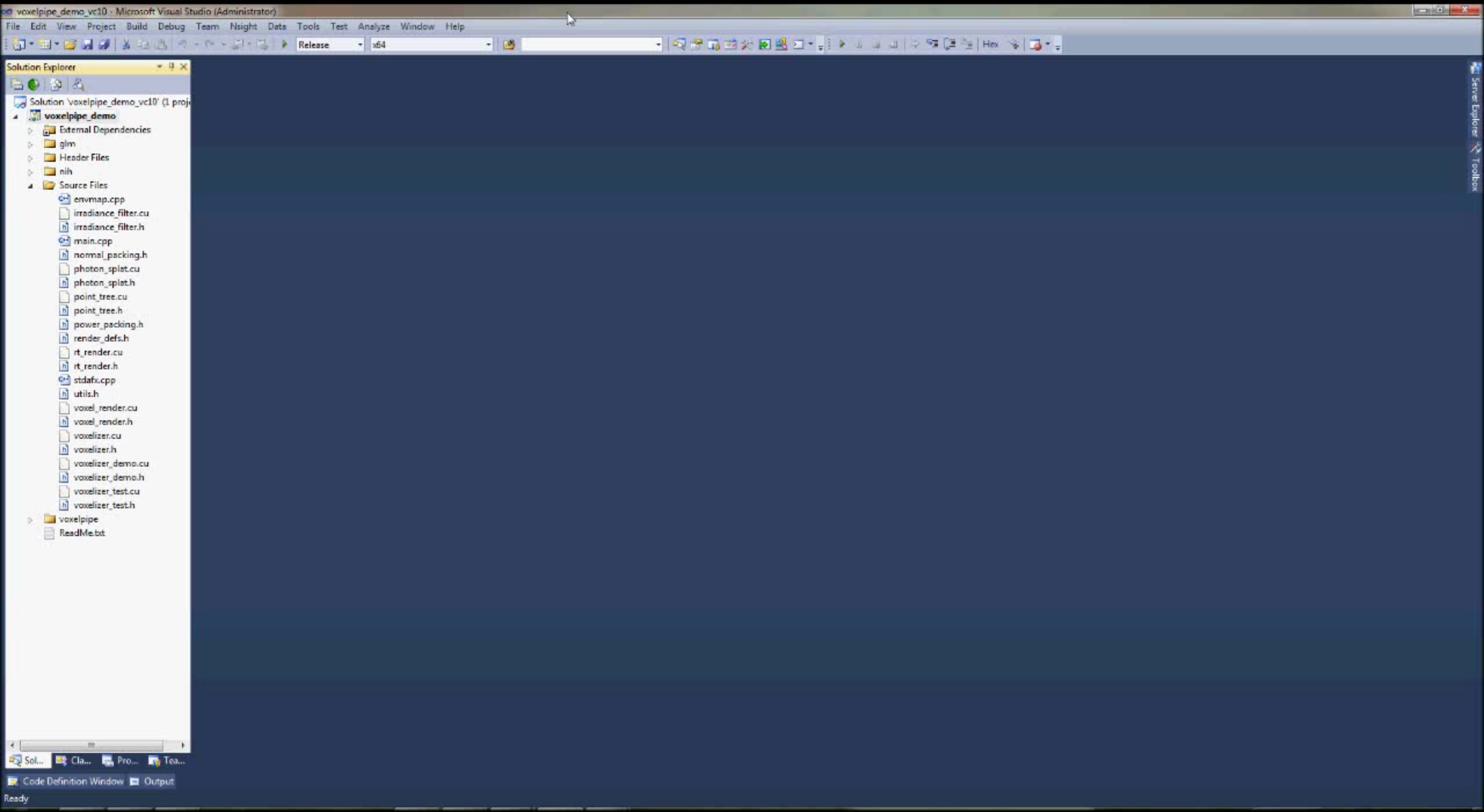
Name	Total	Per Warp	Per Second
Global			
Requests	625,000.00	325.52	250,765,700.00
Loads	375,000.00	195.31	150,459,400.00
Stores	250,000.00	130.21	100,306,300.00
Transactions	625,000.00	325.52	250,765,700.00
Loads	375,000.00	195.31	150,459,400.00
Stores	250,000.00	130.21	100,306,300.00
Size	76.29 MB	40.69 kB	29.89 GB/s
Loads	45.78 MB	24.41 kB	17.94 GB/s
Stores	30.52 MB	16.28 kB	11.96 GB/s
Replay Overhead	0.46 %		
Local			
Requests	6,000,000.00	3,125.00	2,407,351,000.00
Loads	3,000,000.00	1,562.50	1,203,676,000.00
Stores	3,000,000.00	1,562.50	1,203,676,000.00

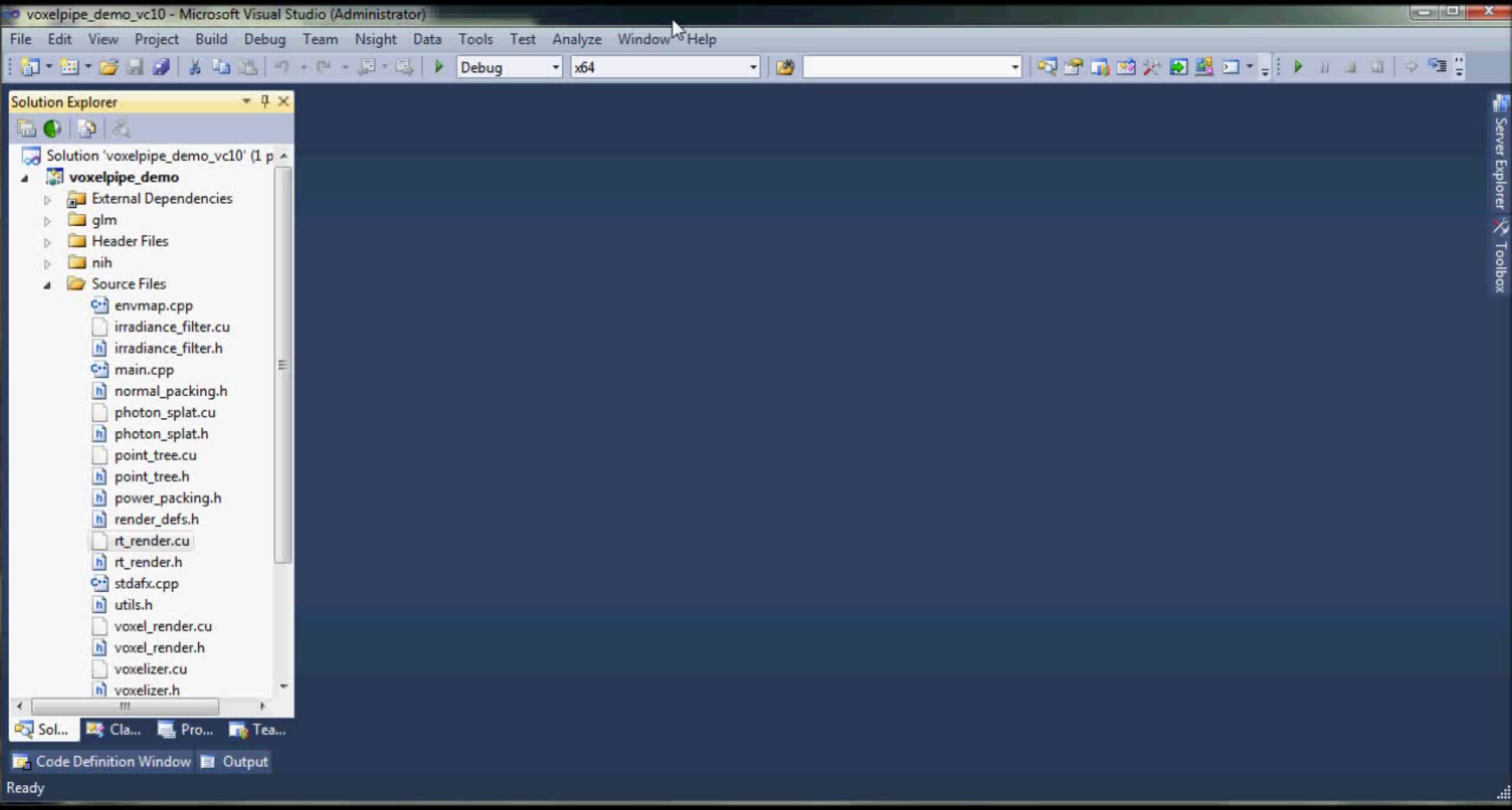
Overview | Global | Local | Atomics | Shared | Texture | Caches | Buffers

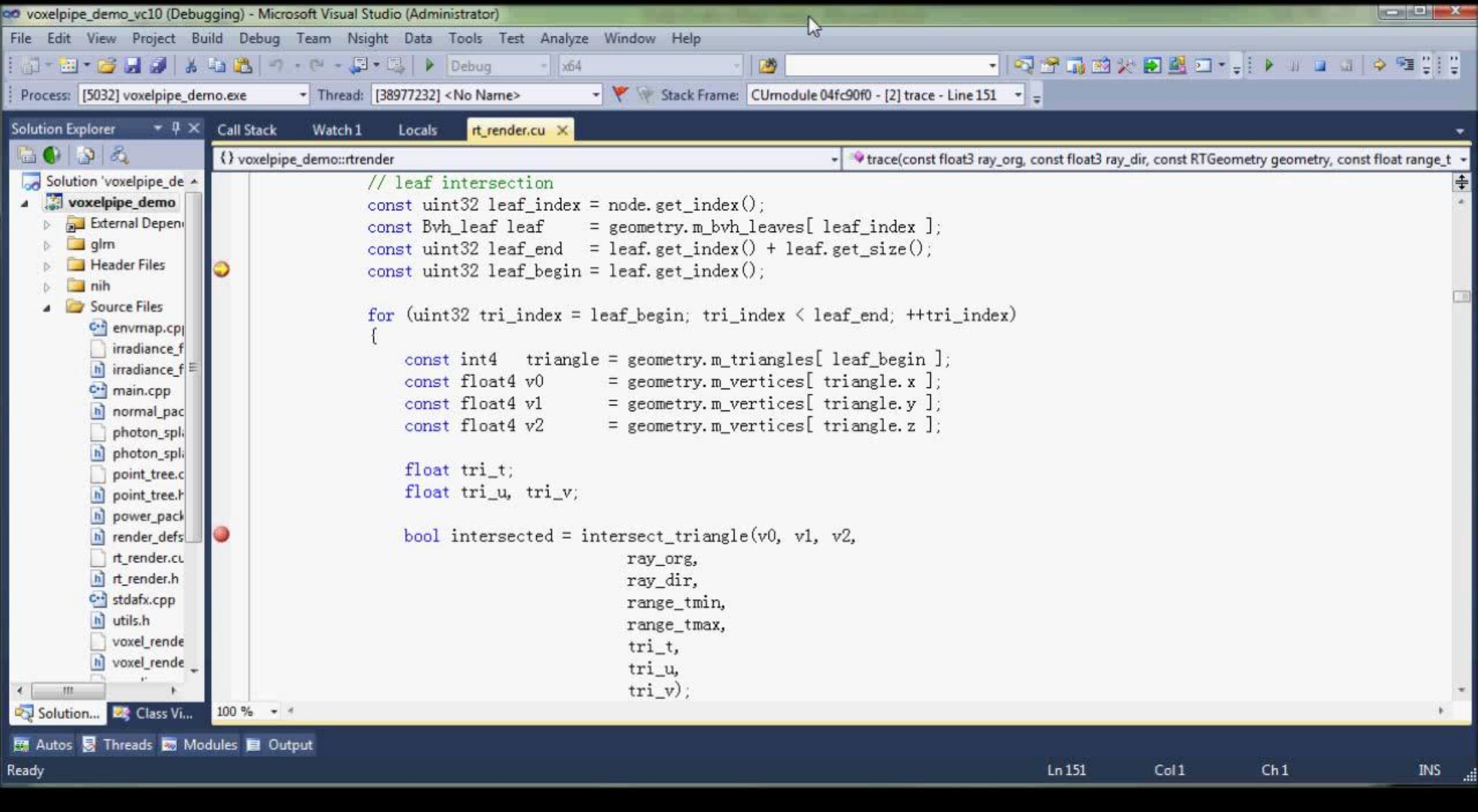
Demo - CUDA

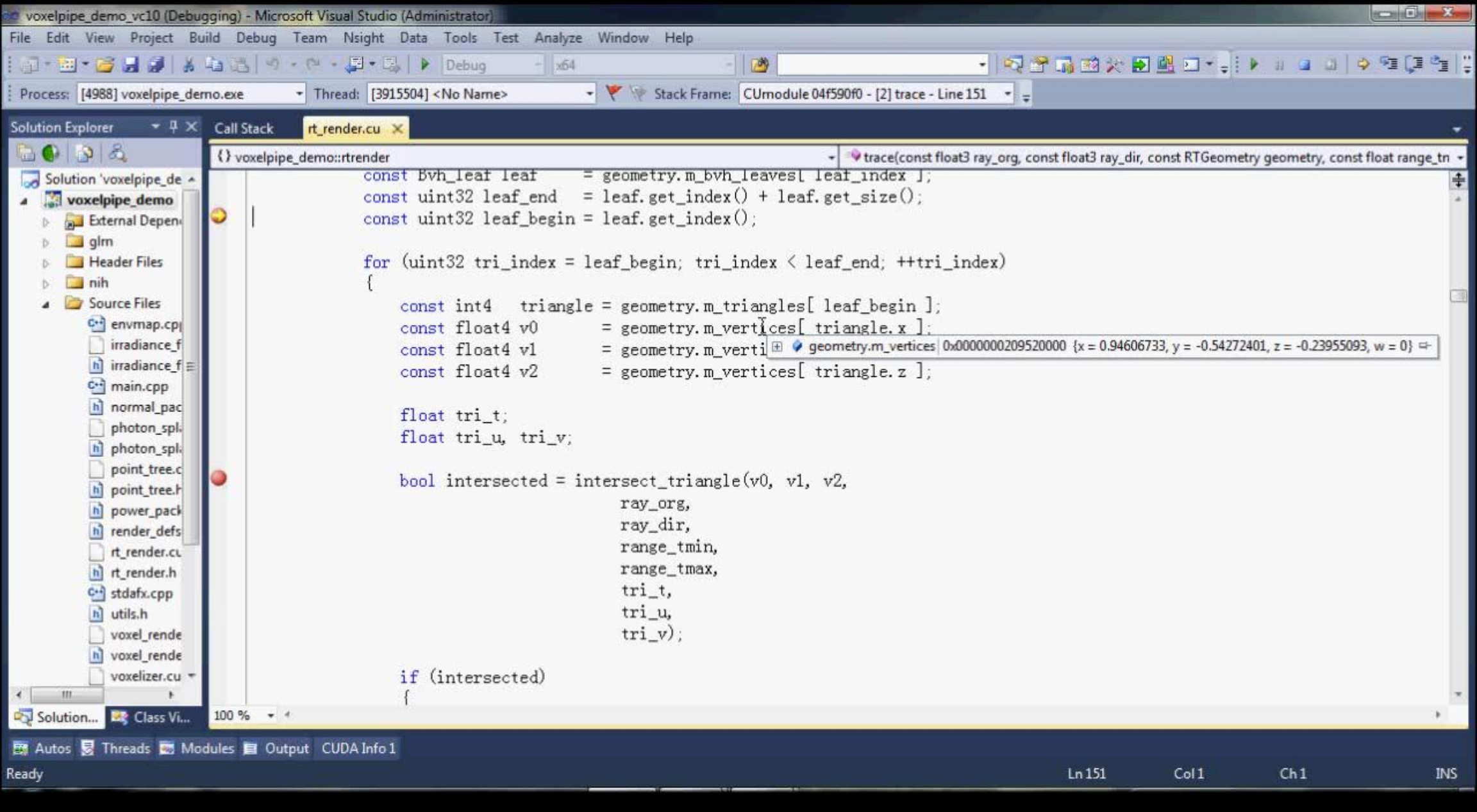
- Similar experience as CPU debugging
- Debug the ‘parallel’
- PTX/SASS
- Memory

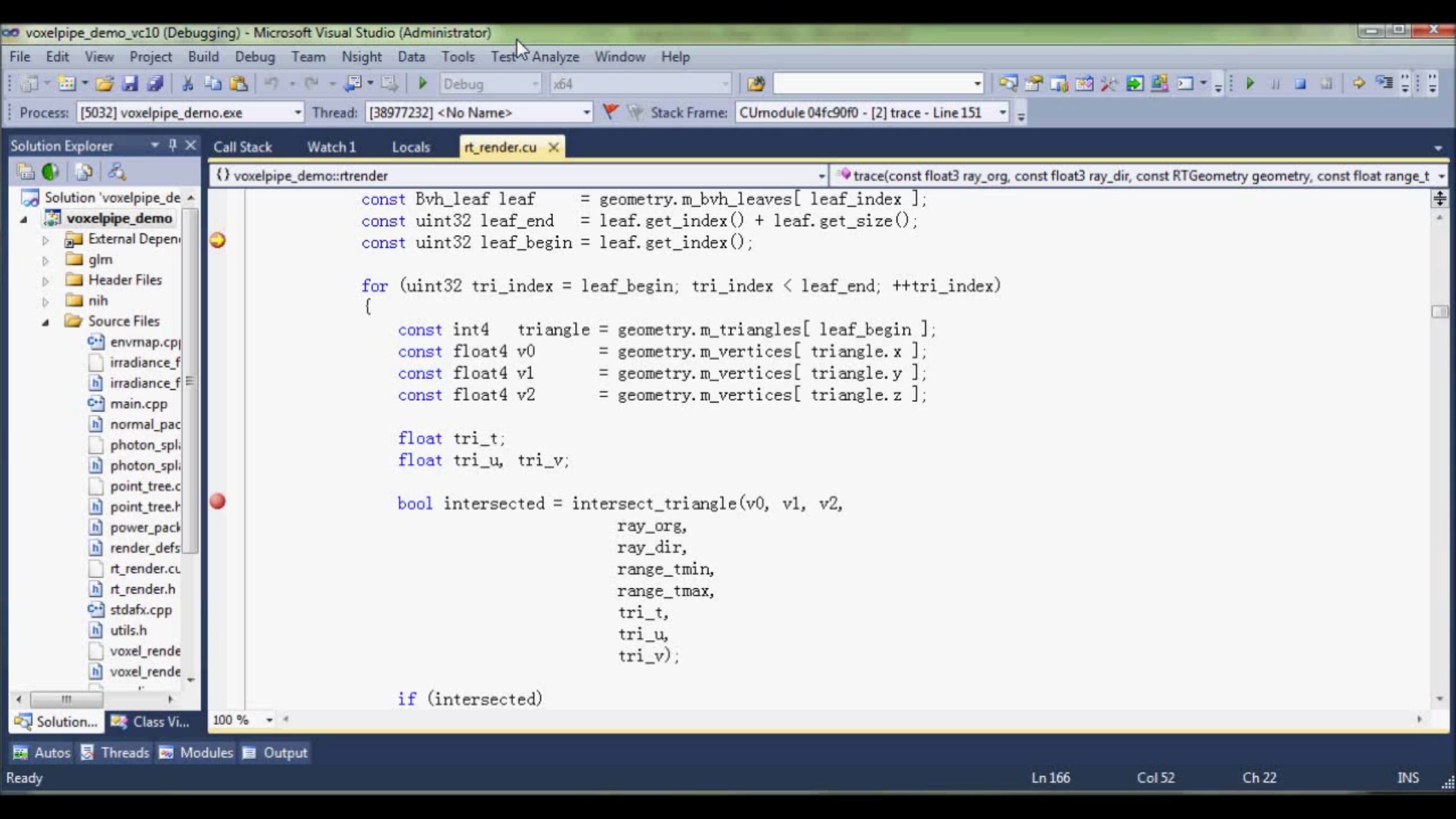


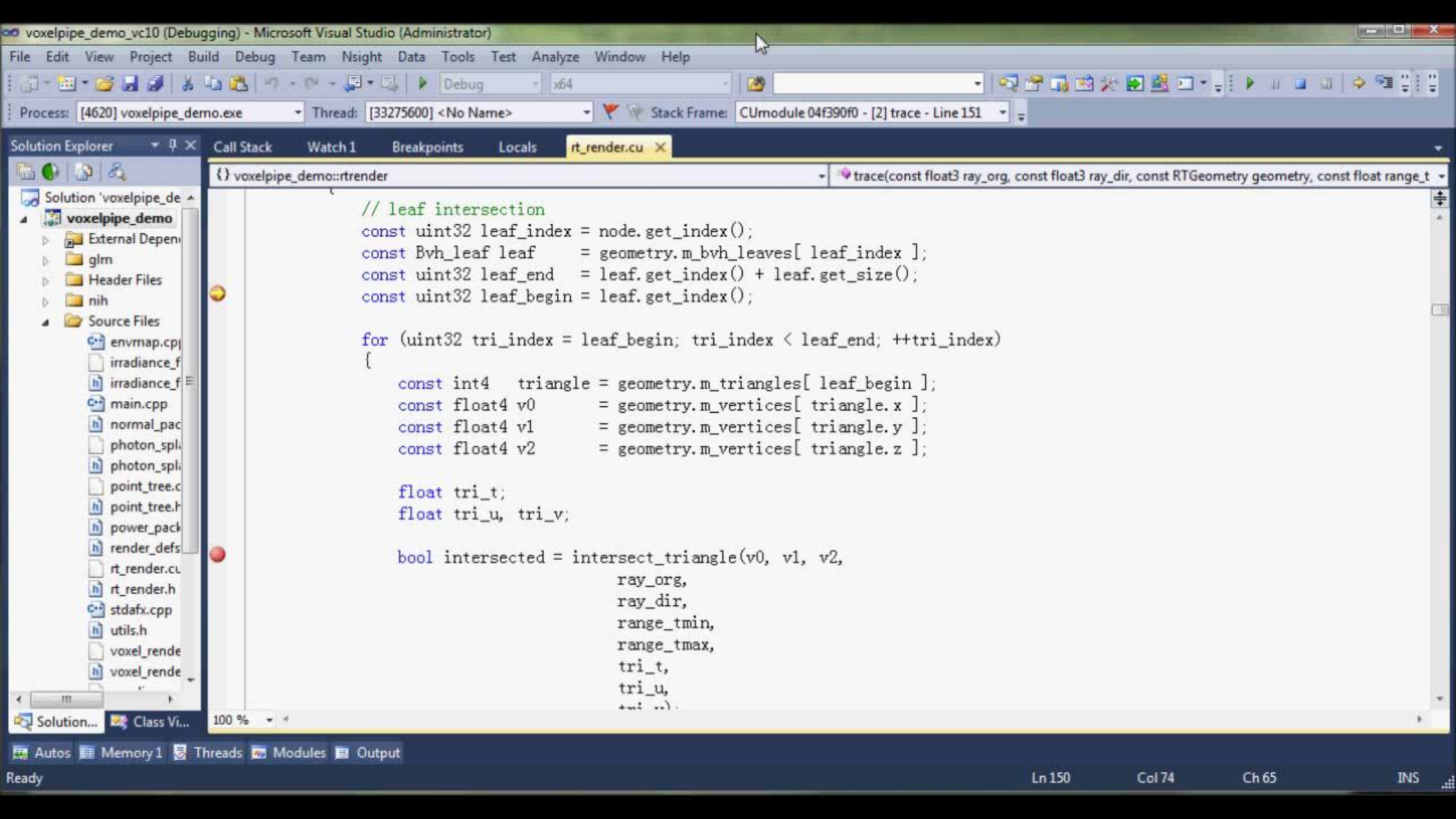

voxelpipe_demo_vc10 - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Analyze Window Help
Release
x64
Solution Explorer
Solution 'voxelpipe_demo_vc10' (1 proj
voxelpipe_demo
External Dependencies
glm
Header Files
nih
Source Files
envmap.cpp
irradiance_filter.cu
irradiance_filter.h
main.cpp
normal_packing.h
photon_splat.cu
photon_splat.h
point_tree.cu
point_tree.h
power_packing.h
render_defs.h
rt_render.cu
rt_render.h
stdafx.cpp
utils.h
voxel_render.cu
voxel_render.h
voxelizer.cu
voxelizer.h
voxelizer_demo.cu
voxelizer_demo.h
voxelizer_test.cu
voxelizer_test.h
voxelpipe
ReadMe.txt
Sol...
Cla...
Pro...
Tea...
Code Definition Window
Output
Ready
Server Explorer
Toolbox


voxelpipe_demo_vc10 - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Analyze Window Help
Debug
x64
Solution Explorer
Solution 'voxelpipe_demo_vc10' (1 p
voxelpipe_demo
External Dependencies
glm
Header Files
nih
Source Files
envmap.cpp
irradiance_filter.cu
irradiance_filter.h
main.cpp
normal_packing.h
photon_splat.cu
photon_splat.h
point_tree.cu
point_tree.h
power_packing.h
render_defs.h
rt_render.cu
rt_render.h
stdafx.cpp
utils.h
voxel_render.cu
voxel_render.h
voxelizer.cu
voxelizer.h
Sol...
Cla...
Pro...
Tea...
Code Definition Window
Output
Ready
Server Explorer
Toolbox

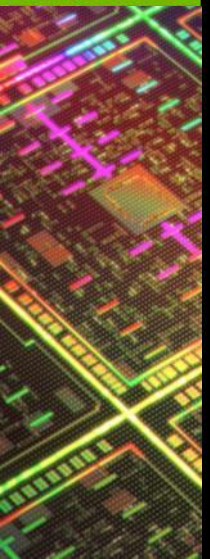

voxelpipe_demo_vc10 (Debugging) - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Analyze Window Help
Debug x64
Process: [5032] voxelpipe_demo.exe Thread: [38977232] <No Name> Stack Frame: CUmodule 04fc90f0 - [2] trace - Line 151
Solution Explorer
Call Stack Watch 1 Locals rt_render.cu
voxelpipe_demo::rtrender
trace(const float3 ray_org, const float3 ray_dir, const RTGeometry geometry, const float range_t
Solution 'voxelpipe_de
voxelpipe_demo
External Depen
glm
Header Files
nih
Source Files
envmap.cp
irradiance_f
irradiance_f
main.cpp
normal_pac
photon_spl
photon_spl
point_tree.c
point_tree.h
power_pack
render_defs
rt_render.cu
rt_render.h
stdafx.cpp
utils.h
voxel_rende
voxel_rende
// leaf intersection
const uint32 leaf_index = node.get_index();
const Bvh_leaf leaf = geometry.m_bvh_leaves[leaf_index];
const uint32 leaf_end = leaf.get_index() + leaf.get_size();
const uint32 leaf_begin = leaf.get_index();
for (uint32 tri_index = leaf_begin; tri_index < leaf_end; ++tri_index)
{
const int4 triangle = geometry.m_triangles[leaf_begin];
const float4 v0 = geometry.m_vertices[triangle.x];
const float4 v1 = geometry.m_vertices[triangle.y];
const float4 v2 = geometry.m_vertices[triangle.z];
float tri_t;
float tri_u, tri_v;
bool intersected = intersect_triangle(v0, v1, v2,
ray_org,
ray_dir,
range_tmin,
range_tmax,
tri_t,
tri_u,
tri_v);
100 %
Solution... Class Vi...
Autos Threads Modules Output
Ready
Ln 151 Col 1 Ch 1 INS


voxelpipe_demo_vc10 (Debugging) - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Analyze Window Help
Process: [4988] voxelpipe_demo.exe
Thread: [3915504] <No Name>
Stack Frame: CUmodule 04f590f0 - [2] trace - Line 151
Solution Explorer
Call Stack
rt_render.cu
voxelpipe_demo::rtrender
trace(const float3 ray_org, const float3 ray_dir, const RTGeometry geometry, const float range_tn
const Bvh_leaf leaf = geometry.m_bvh_leaves[leaf_index];
const uint32 leaf_end = leaf.get_index() + leaf.get_size();
const uint32 leaf_begin = leaf.get_index();
for (uint32 tri_index = leaf_begin; tri_index < leaf_end; ++tri_index)
{
const int4 triangle = geometry.m_triangles[leaf_begin];
const float4 v0 = geometry.m_vertices[triangle.x];
const float4 v1 = geometry.m_verti
geometry.m_vertices 0x0000000209520000 {x = 0.94606733, y = -0.54272401, z = -0.23955093, w = 0}
const float4 v2 = geometry.m_vertices[triangle.z];
float tri_t;
float tri_u, tri_v;
bool intersected = intersect_triangle(v0, v1, v2,
ray_org,
ray_dir,
range_tmin,
range_tmax,
tri_t,
tri_u,
tri_v);
if (intersected)
{
100 %
Autos Threads Modules Output CUDA Info 1
Ready
Ln 151 Col 1 Ch 1 INS

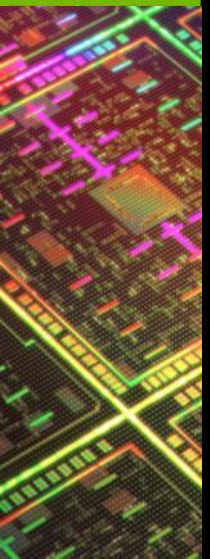

voxelpipe_demo_vc10 (Debugging) - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Analyze Window Help
Debug x64
Process: [5032] voxelpipe_demo.exe Thread: [38977232] <No Name> Stack Frame: CUmodule 04fc90f0 - [2] trace - Line 151
Solution Explorer
Call Stack Watch 1 Locals rt_render.cu
voxelpipe_demo::rtrender
trace(const float3 ray_org, const float3 ray_dir, const RTGeometry geometry, const float range_t
Solution 'voxelpipe_de
voxelpipe_demo
External Depen
glm
Header Files
nih
Source Files
envmap.cp
irradiance_f
irradiance_f
main.cpp
normal_pac
photon_spl
photon_spl
point_tree.c
point_tree.h
power_pack
render_defs
rt_render.cu
rt_render.h
stdafx.cpp
utils.h
voxel_rende
voxel_rende
const Bvh_leaf leaf = geometry.m_bvh_leaves[leaf_index];
const uint32 leaf_end = leaf.get_index() + leaf.get_size();
const uint32 leaf_begin = leaf.get_index();
for (uint32 tri_index = leaf_begin; tri_index < leaf_end; ++tri_index)
{
const int4 triangle = geometry.m_triangles[leaf_begin];
const float4 v0 = geometry.m_vertices[triangle.x];
const float4 v1 = geometry.m_vertices[triangle.y];
const float4 v2 = geometry.m_vertices[triangle.z];
float tri_t;
float tri_u, tri_v;
bool intersected = intersect_triangle(v0, v1, v2,
ray_org,
ray_dir,
range_tmin,
range_tmax,
tri_t,
tri_u,
tri_v);
if (intersected)
100 %
Solution... Class Vi...
Autos Threads Modules Output
Ready
Ln 166 Col 52 Ch 22 INS


voxelpipe_demo_vc10 (Debugging) - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Nsight Data Tools Test Analyze Window Help
Debug x64
Process: [4620] voxelpipe_demo.exe Thread: [33275600] <No Name> Stack Frame: CUmodule 04f390f0 - [2] trace - Line 151
Solution Explorer
Call Stack Watch 1 Breakpoints Locals rt_render.cu
voxelpipe_demo::rtrender
trace(const float3 ray_org, const float3 ray_dir, const RTGeometry geometry, const float range_t
Solution 'voxelpipe_de
voxelpipe_demo
External Depen
glm
Header Files
nih
Source Files
envmap.cp
irradiance_f
irradiance_f
main.cpp
normal_pac
photon_spl
photon_spl
point_tree.c
point_tree.h
power_pack
render_defs
rt_render.cu
rt_render.h
stdafx.cpp
utils.h
voxel_rende
voxel_rende
// leaf intersection
const uint32 leaf_index = node.get_index();
const Bvh_leaf leaf = geometry.m_bvh_leaves[leaf_index];
const uint32 leaf_end = leaf.get_index() + leaf.get_size();
const uint32 leaf_begin = leaf.get_index();
for (uint32 tri_index = leaf_begin; tri_index < leaf_end; ++tri_index)
{
const int4 triangle = geometry.m_triangles[leaf_begin];
const float4 v0 = geometry.m_vertices[triangle.x];
const float4 v1 = geometry.m_vertices[triangle.y];
const float4 v2 = geometry.m_vertices[triangle.z];
float tri_t;
float tri_u, tri_v;
bool intersected = intersect_triangle(v0, v1, v2,
ray_org,
ray_dir,
range_tmin,
range_tmax,
tri_t,
tri_u,
100 %
Solution... Class Vi...
Autos Memory 1 Threads Modules Output
Ready
Ln 150 Col 74 Ch 65 INS

Demo - CUDA Trace



Demo - CUDA Profiling



How can I learn more about Parallel Nsight?

- Download
  - http://parallelnsight.nvidia.com/
- Parallel Nsight documentation
  - Start → All Programs → NVIDIA Parallel Nsight 2.1 → User Guide
- Parallel Nsight instruction videos
  - http://www.gputechconf.com/object/gtc-express-webinar.html
- CUDA books and references
  - Programming Massively Parallel Processors
  - CUDA by Example